

# Indian Computer Science (CS) & Information Technology (IT) Academic Reform (Past) Activism Blog Book

[eklavyasai.blogspot.com](http://eklavyasai.blogspot.com)  
(main work years: 2011 to 2014)

**Ravi S. Iyer  
and others**

Not-for-profit for  
author; Free reuse  
under conditions

**Indian Computer Science (CS) &  
Information Technology (IT)  
Academic Reform (Past) Activism  
Blog Book**

**[eklavyasai.blogspot.com](http://eklavyasai.blogspot.com)  
(main work years: 2011 to 2014)**

**Ravi S. Iyer  
(with some contributions from others)**

**Copyright:** Author Ravi S. Iyer's content which forms most of this book is FREELY REUSABLE (no copyright) but under some conditions. The chapter in this book titled, "Conditional Permission to freely reuse MY CONTENT of this book" gives the details.

**Not-For-Profit:** This book is NOT FOR PROFIT for the author Ravi S. Iyer who is also the self-publisher. But the self-publishing support provider who is not the author (e.g. self-publishing company), printer, distributor and seller may profit.

**Published by:** Ravi S. Iyer (Self-published) in Puttaparthi, Andhra Pradesh, India

**First published:** 2020

**First edition, ebook**

**ISBN:** 978-93-5406-358-9

**Book availability:** At <https://eklavyasai.blogspot.com/p/indian-computer-science-cs-information.html> (short link: <http://bit.ly/indian-cs> ), which has/will have ebook free download and paperback sale page links.

### **Social media work of Ravi S. Iyer**

- About Sri Sathya Sai Baba and more: <http://ravisiyer.blogspot.com>
- Existential reality, Vedanta and God & science conversation: <http://iami1.wordpress.com>
- Indian CS & IT Academic Reform (Past) Activism: <http://eklavyasai.blogspot.com/p/table-of-contents.html>
- Course material related to computer programming (software lab.) courses: <http://raviiyerteaches.wordpress.com>
- Misc. topics: <http://ravisiyermisc.blogspot.com/>

Email: [ravi@raviiyer.org](mailto:ravi@raviiyer.org)

Facebook: <http://facebook.com/ravi.s.iyer.7>

Twitter: <http://twitter.com/RaviSaiIyer>

My motto: Service to Society is Service to God

## *Dedication*

To the field of software development (international),  
which gave me a decent livelihood  
and a decent life

Main author Ravi S. Iyer created the eklavyasai.blogspot.com blog and used it from September 2011 to play a part-time, peaceful and amicable, Indian Computer Science (CS) and Information Technology (IT) academic reform, Internet-based activist role. His focus was on improving the practice of software development in Indian CS & IT academia. But he thought that it is such a vital part of the CS & IT field and that it is so poor in many parts of Indian CS & IT academia, that he referred to his efforts as Indian CS & IT academic reform activism.

Other contributors to the blog have given their views on certain topics. Main work period has been from 2011 to 2014 with a little work later, off & on. The main author is no longer active in this area. This book is aimed at helping other activists involved in improving the practice of software development in Indian CS and IT academia to get the views of the blog in a convenient form. The book may also be of interest to similar activists in other countries.

Main author Ravi S. Iyer is a Physics graduate from Ruia college, University of Bombay (Mumbai) who was industry trained and later self-taught in software development. He worked in the international software industry (US, Europe, Japan, South Korea, India etc.) developing systems as well as applications software (CS & IT) for over 18 years after which he retired from commercial work. Later, mainly as a "visiting faculty", he offered free service of teaching programming courses (lab. courses) and being a "technical consultant" for student projects in a Maths & Computer Science department of a deemed university in India for 9 years.

For at cost or minimum price printed book and free ebook visit <http://bit.ly/indian-cs>

## Contents

Conditional Permission to freely reuse MY CONTENT of this book .....	<b>10</b>
Preface.....	<b>11</b>
Truth Telling - A Tough Job.....	<b>14</b>
 <b><i>Improving the Teaching and Practice of Software Development in Indian CS &amp; IT Academia</i></b>	
My preprint "Paper", Improve the Practice of Software Development in India by ...: Ranked highly by Internet search engines and Google Scholar; Paper text.....	<b>16</b>
Ten Computer Science (CS) & Information Technology (IT) higher education policy changes to improve practice of software development in India .....	<b>45</b>
Concrete Suggestions for Measuring Teaching Quality in Practice-Oriented Computer Science/Information Technology streams.....	<b>46</b>
Discussion on Concrete Suggestions for Measuring Teaching Quality ... ..	<b>49</b>
Wrote to Prime Minister, Shri Narendra Modi, on improving practice of software development in Indian CS & IT academia .....	<b>51</b>
Poor software development skills Indian CS graduates - Google Search Results.....	<b>52</b>
Comments (of former CS academic from UK) on Software Development Career Track Paper .....	<b>55</b>
CS & IT Academia: Serious Systemic Problems? .....	<b>56</b>
CS & IT Academia: How To Relieve Suffering of Students, Parents and Employers.....	<b>63</b>
CS & IT Academia: Inform Students About Software Contribution Record of Faculty.....	<b>64</b>
CS & IT Academia: Profs. May Not Like Idea of Software Contribution Record.....	<b>66</b>
Publicly Funded Higher Education Institutions should put up Detailed Course contents on the Internet.....	<b>67</b>
My suggestions related to skill development in Indian higher education on mygov.in discussions on New Education Policy.....	<b>69</b>
How MHRD (& DIRECTOR IIT) disposed of my "Serious Systemic Problems in Indian CS IT Academia" grievance dated Nov. 2011, DOING NOTHING REALLY .....	<b>71</b>
About me not being Ph.D. or qualified in Computer Science but yet teaching lab. courses and being tech. consultant in an Indian deemed university in the past.....	<b>72</b>

## ***Views of Top Leaders (Country Presidents/Prime Minister/Ministers/US Senators) on Education***

Education and IT related parts of President Mukherjee's address to Parliament; My comments .....	<b>76</b>
Education and IT related excerpts from PM Narendra Modi speech in Lok Sabha on June 11th 2014.....	<b>77</b>
Education Related Parts of US President Obama's State of the Union 2014 Speech .....	<b>81</b>
USA - Student Right to Know Before You Go Act.....	<b>82</b>
First Question in 2nd US Presidential debate: Student asks about employment after graduation! .....	<b>84</b>

## ***Research vs. Teaching***

Nature and Science (Top International Research Magazines) 2011 Articles on Lack of Importance Given to Teaching Nowadays .....	<b>86</b>
Suggestion of Separate Tracks for Teaching and Research by US Academics.....	<b>86</b>
India (and Other Emerging/Developing Countries) Should Not Obsess about Higher Education Rankings .....	<b>87</b>
Improving Indian Academic Research and Teaching: Have Separate Research-Intensive Universities and Teaching-Intensive Universities .....	<b>89</b>
Is Academic Research Grant Money Corrupting Academic Teaching Ideals? .....	<b>91</b>
CS & IT Academia: Is Teaching Excellence Important? .....	<b>93</b>
Is a PhD in CS/IT Necessarily a Good Teacher?.....	<b>96</b>
A Defense of The CS/IT PhD Teacher.....	<b>99</b>
Suggestion of Separate University Rankings for Research Excellence and Teaching Excellence .....	<b>101</b>
CS & IT Academia: Research vs. Teaching.....	<b>104</b>
Should M.Tech.(CS) Project be CS Research Oriented Or Software Engineering Project Oriented? .....	<b>107</b>
HRD Minister Javadekar says research not necessary for promotion for college teachers; university dept. faculty however need to do research .....	<b>118</b>
HRD minister Javadekar tells truth about UGC AICTE academia problems; removes mandatory research for college faculty (but not university faculty); assures UGC AICTE reform.....	<b>119</b>

Dr. Pushkar's articles on research req. for Indian college teachers producing fake research and reducing teaching quality; historical background of Indian academic research system ..... **127**

### ***IT Finishing Schools***

IT Finishing Schools..... **131**

More on IT Finishing Schools ..... **133**

### ***Academic PhD Club & Bureaucratic POWER structure***

CS & IT Academia: The PhD Glass Ceiling..... **136**

CS & IT Academia: A Bureaucratic POWER structure ..... **137**

The very strange case of 20th century era Department(s) of Mathematics and Computer Science in Indian academia in today's early 21st century world ..... **139**

### ***Indian Engineering Colleges: Graduate Employability Problems***

Private study claims: Less than 9% of Indian engineering students have required programming and algorithm skills for IT product company jobs..... **142**

GATE and UGC NET CS & IT exams can be employability measures if they include practical knowledge assessment..... **144**

The Hindu's higher education student guidebook - thenxt.step 2013 - CS & IT picture . **144**

2014 National (India) Employability report - Poor Employability of Andhra Pradesh Engineering Graduates..... **148**

### ***Miscellaneous***

A Tough View of UK CS Academia ..... **151**

The Software Development Professor! ..... **152**

Is (Seoul Accord) Accreditation using Outcome Based Assessment methods the way forward to improve teaching standards in Indian CS & IT academia? ..... **155**

US CS PhD student's Comparative View of US and Indian Academia ..... **160**

Prof. Stroustrup: Software development is a potentially noble profession like medicine or classical engg. disciplines but long way to go ..... **164**



## ***Appendices***

Appendix A: My software field biodata, software industry work experience and lab. courses I taught .....	<b>167</b>
Appendix B: Holistically Elite CS/IT Education vs. Technically Elite CS/IT Education ...	<b>173</b>
Appendix C: Discussion and Debate on lack of Proper Warranty for Software, and need thereof .....	<b>175</b>
Appendix D: List of some other eklavyasai blog posts including on “How to Teach Programming?” .....	<b>186</b>

# **Conditional Permission to freely reuse MY CONTENT of this book**

1. I hereby grant permission to anybody, anywhere to freely reuse (including adaptation) any of MY CONTENT of this book, provided they give a reference to this book to enable interested persons to easily cross-check with the original source, and provided they accept the other points listed below in this Permission section. An example of such reuse would be somebody editing part (or whole) of this book and then publishing and printing it as their (derived) work. Such reuse (including adaptation) of any of MY CONTENT of this book need not be limited to NOT-FOR-PROFIT work – it can be done for FOR-PROFIT work like publishing & printing a book for profit. I do NOT WANT any royalty or profit from any such FOR-PROFIT work reusing MY CONTENT of this book.
2. Any legal challenges/lawsuits filed/made by others (NOT me) related to reuse of MY CONTENT of this book, against people who did such reuse of MY CONTENT of this book, has to be handled by them (people who reused MY CONTENT) without any legal liability to me, except as required by Indian law.
3. If at all somebody is interested in translating any or all of MY CONTENT of this book to some other language then that is just another case of the reuse that I have mentioned above in point 1. So anybody is free (I grant them permission) to do such translation of MY CONTENT of this book and publish & print it, even FOR-PROFIT for them. I do NOT WANT any of the profits they may make from this activity.
4. If at all somebody is interested in printing and selling any or all of MY CONTENT of this book without any changes, then that too is just another case of the reuse that I have mentioned above. So anybody is free (I grant them permission) to do such printing and selling of MY CONTENT of this book, even FOR-PROFIT for them. I do NOT WANT any of the profits they may make from this activity.
5. Note that this permission applies only to MY CONTENT in this book and DOES NOT APPLY to small amount of content I may have used from elsewhere which I would have made every attempt to clearly show as having been taken from elsewhere or from somebody else. If any content of this book is not shown as having been taken from elsewhere or from somebody else, then it can be presumed that that content is MY CONTENT.

28<sup>th</sup> February 2020

Ravi S. Iyer

Puttaparthi, Andhra Pradesh, India

# Preface

This blog book and associated eklavyasai.blogspot.com blog are part of my humble give-back contributions in gratitude to the field of software development where I earned a decent livelihood, helped fulfill my family obligations which was made possible by my software profession earnings and enjoyed my work most of the time. I was in the software industry from March 1984 to Aug. 2002 joining it as a trainee programmer, progressing over the years to Programmer/Analyst (software developer), Project Leader, Project Manager and Software Manager finally in 1990 (all positions were as employee of software companies). From around March 1993, I became a freelance corporate trainer in software development (programming courses) and later an independent technical consultant (no longer an employee) which lasted till I took commercial retirement from the field in Aug. 2002.

The software development (SD) field enabled me to make great friendships with fellow software developers, and others like support staff and managers in software companies that I worked in or interacted with in India as well as outside India, get exposed to life in USA and Europe over assignment stints there totaling around 3 years, and get interesting exposure to South Korea through a short around 2 month stint there. My SD field work also involved interesting marketing trips to Europe and Singapore. It enabled me to retire from commercial work at 40 years of age to pursue my spiritual aspirations while leading a simple, single (unmarried) spiritual aspirant life.

The logical analysis skills which I had learned in school and college (I am a Physics graduate), were honed by the SD field through analysis of complex programs and large software systems. Software design and architecture to create technically sound software systems were taught to me by the SD field. It taught me that being honest about mistakes made by me, some of which were pointed out by peers and customers, rather than go into arrogant denial, was the way to produce better software designs and code. I learned that in software code and design, arrogant denial does not overcome truth, even if denial may cloud the truth for some time. Eventually it is the truth that triumphs over falsehood including arrogant denial of truth. The computer machine as well as the SD field in general is awesome in how they do not tolerate falsehood and how truth always triumphs on the computer machine and in the SD field in general. I learned to overcome the egoistic discomfort I initially felt when design reviewers pointed out flaws and suggested improvements in my designs, and matured as a software designer to the stage where I humbly and wholeheartedly welcomed and indeed, craved, for suggestions for improvement as well as flaws being pointed out, in my designs by peers.

Handling people, understanding their interests and views, motivating them, giving them reasonable freedom in their work, and cherishing and celebrating team achievements were taught to me both by my SD field management work and also by learning how managers that I reported to, and my peers handled me and my many human flaws. The great amount of literature and documentation that I read while in the SD field on technical as well as management and business related matters played a vital role in significantly expanding the horizons of my mind. It is the SD field that made me grow as a person from a Physics graduate job-seeker to a well settled and confident man who either knew his limitations or, especially in initial stages of my SD field career, was brutally made to realize such limitations by some mistakes and initial failures, and worked within those limitations to deliver on SD field work. Later in life, this growth as a person positioned me to seriously look at and write on larger challenges of understanding ethical/moral norms as well as existential reality of life from an individual perspective and from a larger community perspective, and the role of spirituality & religion as well as rule of law (of the state/country) in those areas, in our times and also in the past.

My bio-data and software industry work experience are given in an appendix of this book. My self-published book titled, "Autobiography of an Indian Software Techie and Spiritual Aspirant – Part 1" and sub-titled "Covers Mumbai & Dombivli based life with some foreign stints, till 2002 (age 40)" covers my software industry work period in a hastily written manner (due to lack of time; my apologies to readers).

<https://ravisiyer.blogspot.com/p/autobiography-of-indian-software-techie.html> [short link: <http://bit.ly/2GVctBG>] has the links for free download of ebook version of above (part autobiography) book as well as paperback sale page links.

I take this opportunity to express my deep gratitude to all the individuals, groups of individuals, companies and organizations including educational & research institutions that created, sustained and developed the software development field. This includes computer scientists, technologists (architects, designers, developers, quality assurance engineers etc.), teachers (including corporate trainers), (business) entrepreneurs, investors and, last but not the least, users.

Another aspect of my humble give-back contributions to the software development field was my stint in a Department of Mathematics & Computer Science in a deemed university in South India, where as free service rendered as a spiritual offering, I taught software lab. courses (software development/programming courses) mainly to Masters students in Mathematics and Computer Science fields, and guided student project work related to software development at Masters level (M.Sc. Mathematics and M.Tech. Computer Science) for around 9 years from Jan. 2003 to Mar. 2012. During this period I had designations of Honorary Staff, Honorary Faculty and Visiting Faculty. I also additionally played a partial Lab. Manager role from Jan. 2003 to around Jan. 2008. I should also mention that I have taught a few software lab. courses to Physics Masters students in this deemed university.

An appendix in this book gives details of the software lab. courses I taught in this deemed university.

The [eklavyasai.blogspot.com](http://eklavyasai.blogspot.com) blog grew out of my exposure to the challenges faced in Indian Computer Science (CS) and Information Technology (IT) academia in teaching software development well to its students.

A few friends have made significant contributions to the [eklavyasai.blogspot.com](http://eklavyasai.blogspot.com) blog which is the basis for this book. Thanks guys. The [eklavyasai](http://eklavyasai.blogspot.com) blog lists the contributors here: <https://eklavyasai.blogspot.com/p/contributors.html>.

The contributors/friends who agreed to have their pen-name/name mentioned in this book are:

1. Anakin (pen name)
2. Vaddadi Kartick – (pen name used in [eklavyasai](http://eklavyasai.blogspot.com) blog: Picard)

The contributors usually do not have their name/pen name mentioned in the blog posts and associated parts of this book, but are named in a few articles. They are usually referred to as simply friend (or correspondent). Any content in the book that is not shown as having been contributed by somebody else, can be presumed to be written by me.

Besides the above mentioned contributors, some other email correspondents, which include some leading international Computer Science academics & industry figures, have also contributed to the [eklavyasai](http://eklavyasai.blogspot.com) blog by agreeing to me sharing some of their email responses on this blog. However, I have included only some of their comments in this book due to permission issues. But I will be referring to those comments which I have not included in the book by directing readers to the associated blog post.

One particular international software industry as well as Computer Science academic person, spent a lot of his valuable time in sharing his views on the matters raised in [eklavyasai](http://eklavyasai.blogspot.com) blog, and gave great encouragement to me to do that work. He prefers to be anonymous and so I am not mentioning his name here. I thank him profusely for the views he shared with me on [eklavyasai](http://eklavyasai.blogspot.com) blog posts and related matters, and for the encouragement he gave me.

Readers are free to disagree with the views expressed in this book and associated blog. I and other contributors have expressed our views (at that time). If some of those views are helpful, or in future become helpful to some readers then the effort put in by me on the blog as well as this book would have had positive

impact. However, even if that does not happen, that's fine by me. I am very satisfied that I have shared my views and thoughts and suggestions, along with those of other contributors, on the important issue of improving the teaching & practice of software development in Indian Computer Science (CS) and Information Technology (IT) academia/higher education, and through that I feel as if I have done my duty to my beloved field of software development which played a big role in making me what I am today. Note that some of the views, thoughts and suggestions in this book may apply to other countries' CS & IT academia/higher education.

Ravi S. Iyer

1st March 2020

# Truth Telling - A Tough Job

Associated blog post date: 2nd Oct. 2012, link: <http://eklavyasai.blogspot.com/2012/10/truth-telling-tough-job.html> , short link: <http://bit.ly/truth-t>

Trying to be a truth-teller in today's world is a tough job. It is an unpopular affair.

I recently saw a few videos and read articles about how the great physicist Feynman faced the same challenges when he investigated the Challenger disaster. It was an eye-opener to me that even such a world-famous physicist had to face significant resistance from powerful administrators. If you have not seen it I recommend you see this 4 min 42 sec. video, Richard Feynman - Space Shuttle Challenger Investigation, [archived link:

<https://web.archive.org/web/20120225062425/http://www.youtube.com/watch?v=UCLgRyKvfp0> (short link: <http://bit.ly/feynman-ch> ) for following broken link]

<http://www.youtube.com/watch?v=UCLgRyKvfp0>. The official view now seems that Feynman did catch the real problem: [http://en.wikipedia.org/wiki/Richard\\_Feynman#Challenger\\_disaster](http://en.wikipedia.org/wiki/Richard_Feynman#Challenger_disaster).

The wiki page above states the following (He is Richard Feynman):

--- start extract ---

He concluded that the space shuttle reliability estimate by NASA management was fantastically unrealistic, and he was particularly angered that NASA used these figures to recruit Christa McAuliffe into the Teacher-in-Space program. He warned in his appendix to the commission's report (which was included only after he threatened not to sign the report), "For a successful technology, reality must take precedence over public relations, for nature cannot be fooled." [174]

...

[Wiki Ref:]

174. Richard Feynman. "Appendix F – Personal observations on the reliability of the Shuttle", <https://history.nasa.gov/rogersrep/v2appf.htm>. Kennedy Space Center. Retrieved September 11, 2017.

--- end wiki extract ---

My respect for the great physicist Richard Feynman went up enormously after I recently came to know of the above human goodness side and the sheer \*guts\* to speak out the truth in the face of powerful opposition side of him.

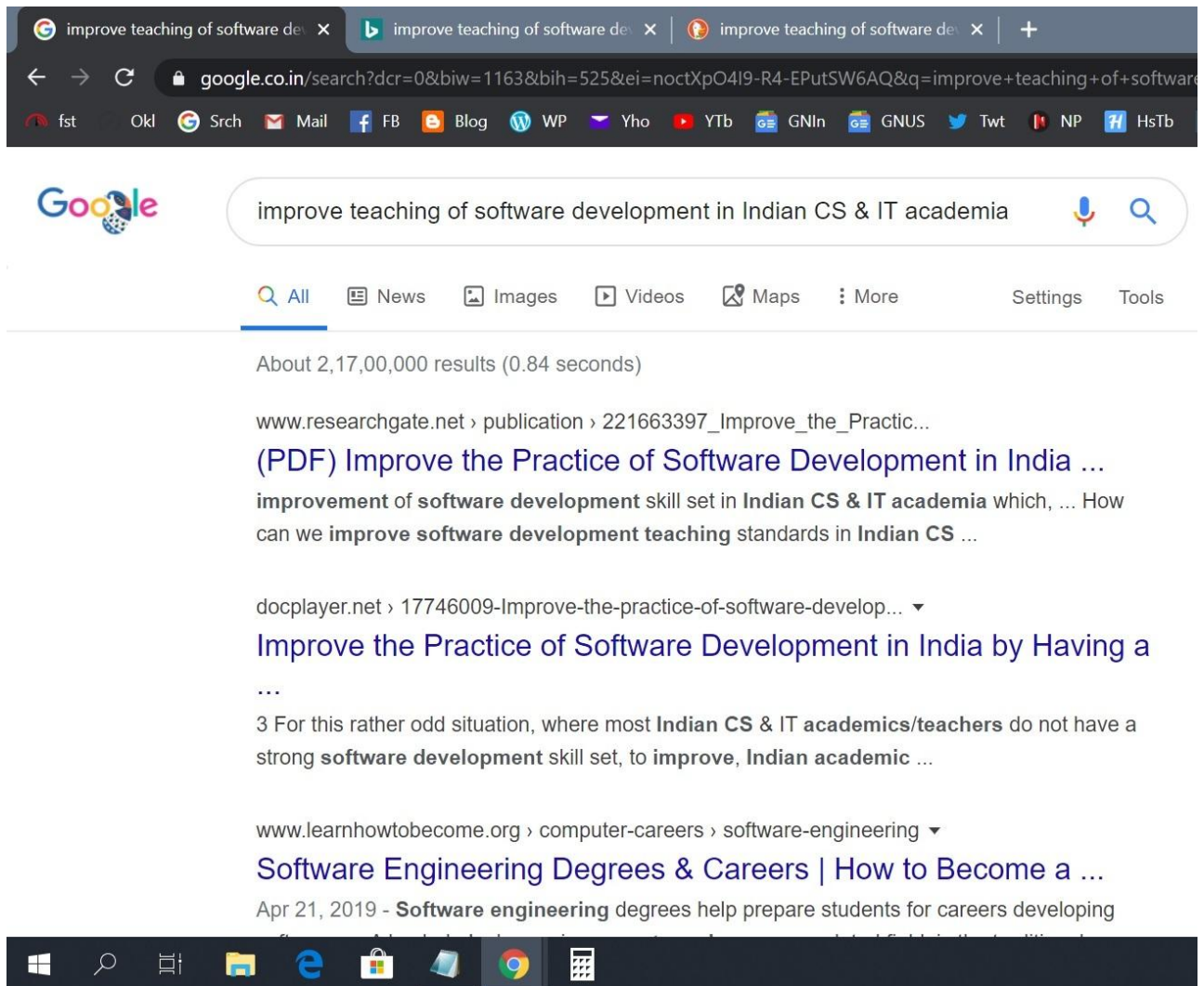
So, I guess, it is always a challenge to be a truth-teller - material truth-teller - and far more challenging perhaps to be a spiritual truth-teller (e.g. [http://en.wikipedia.org/wiki/Mansur\\_Al-Hallaj](http://en.wikipedia.org/wiki/Mansur_Al-Hallaj)). The saying goes: Sathyam Bhruyath Priyam Bhruyath Na Bhruyath Sathyam Apriyam. [English translation: Speak the truth; speak the pleasant truth; don't speak the unpleasant truth.]

Perhaps the safe path is to just put out one's views on the Internet and provide opinions only if people ask - a low-key activism and not a pushy activism.

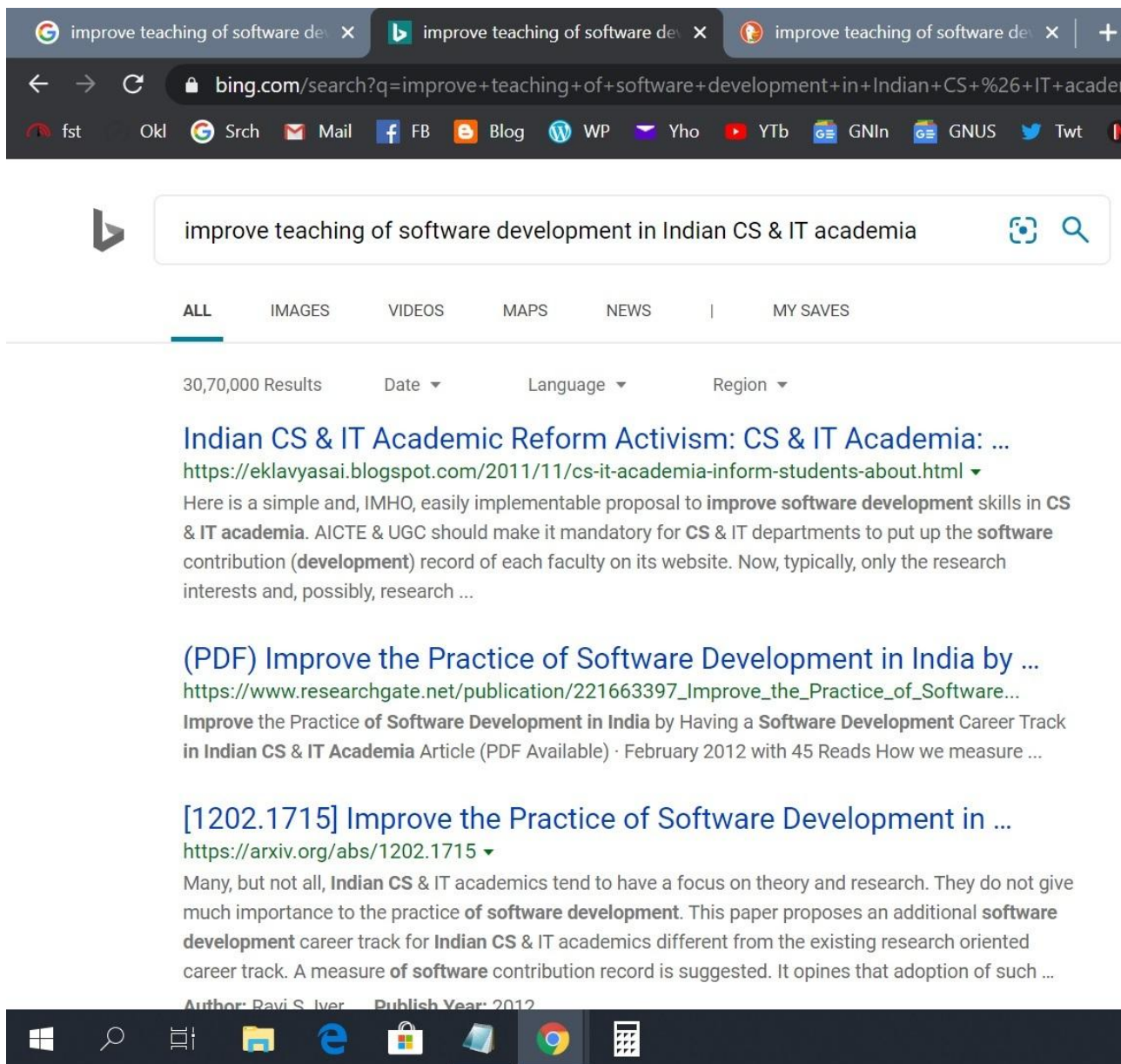
# **Improving the Teaching and Practice of Software Development in Indian CS & IT Academia**

# My preprint "Paper", Improve the Practice of Software Development in India by ...: Ranked highly by Internet search engines and Google Scholar; Paper text

My preprint paper (not published in peer reviewed academic publication) titled, "Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia" which was last updated in December 2012 seems to be ranked highly in Internet Search engines for appropriate terms even in Jan. 2020! Cropped screenshots of some related Google and Bing searches are given below.







The full screenshots and some more information can be seen in my blog post: Quite amazed that Internet search on: improve teaching of software development in Indian CS & IT academia, gives my Dec. 2012 paper as 1st or 2nd result even now in Jan 2020, <https://eklavyasai.blogspot.com/2020/01/quite-amazed-that-internet-search-on.html> [short link: <http://bit.ly/36xcc1r> ], 26th Jan. 2020.

The paper itself is available on multiple sites with the main site being arxiv.org. The related links are given below.

Abstract: <http://arxiv.org/abs/1202.1715>

PDF: <http://arxiv.org/pdf/1202.1715>

The paper contents are given later on in this book (taken from above PDF link).

The word skills has become an important keyword in the context of discourse on employability of graduates in India. So I felt it is important to see where my paper stands in search results having the term skills in the search string.

The cropped screenshots below (from incognito Chrome window to avoid bias towards my stuff) show related searches on Google and Bing search engines rank my paper within the top 3 results.



how to improve software development skills in Indian CS IT academia



All News Videos Images Shopping More Settings Tools

About 99,80,000 results (0.91 seconds)

### The reason India does not have enough awesome developers ...

<https://www.hackerearth.com> › developers › indian-software-developers ▼

Jan 21, 2019 - So why are Indian software developers considered bad? ... hiring at scale has been quite straightforward; shortlisting on the basis of academic performances, a FizzBuzz test ... These developers are on the platform to hone their skills and are keen learners. ... 7 steps to improve your data structure and algo.

### People also ask

- Can I be a programmer with an IT degree? ▼
- How can I become a software developer at home? ▼
- Can I become software engineer without degree? ▼
- How many years can a software engineer work? ▼

Feedback

### Can a non-CS person get a cool CS job by learning some software ...

<https://www.quora.com> › Can-a-non-CS-person-get-a-cool-CS-job-by-lear... ▼

16 answers  
Sep 17, 2014 - Can a non-CS person get a cool CS job by learning some software coding and stuff? ... Aman Goel, B.Tech Computer Science and Engineering, Indian Institute of ... with no experience, no relevant academic creds, and maybe half-knowing ... or without a degree in CS) can showcase his/her software engineering skills to a ...

- How can a non-computer science student get a job as a ... 12 answers 5 Aug 2017
- Will you hire a programmer without a degree in India? 25 answers 4 Nov 2014
- What are some career paths for a Computer Science ... 182 answers 1 Feb 2016
- How to prepare myself to be a software engineer at ... 51 answers 8 Aug 2014

More results from [www.quora.com](http://www.quora.com)

### Improve the Practice of Software Development in India by Having a ...

[docplayer.net](http://docplayer.net) › 17746009-Improve-the-practice-of-software-development... ▼

It opines that adoption of such changes to academic regulations will result in significant improvement of software development skill set in Indian CS & IT ...

### Wrote to Prime Minister ... - Indian CS & IT Academic Reform Activism

[eklavayasai.blogspot.com](http://eklavayasai.blogspot.com) › 2014/05 › wrote-to-prime-minister-shri-narendra  
poor programming skills Indian Computer Science graduates improve ... improve programming skills Indian software graduates --- end comments --- A little while ...

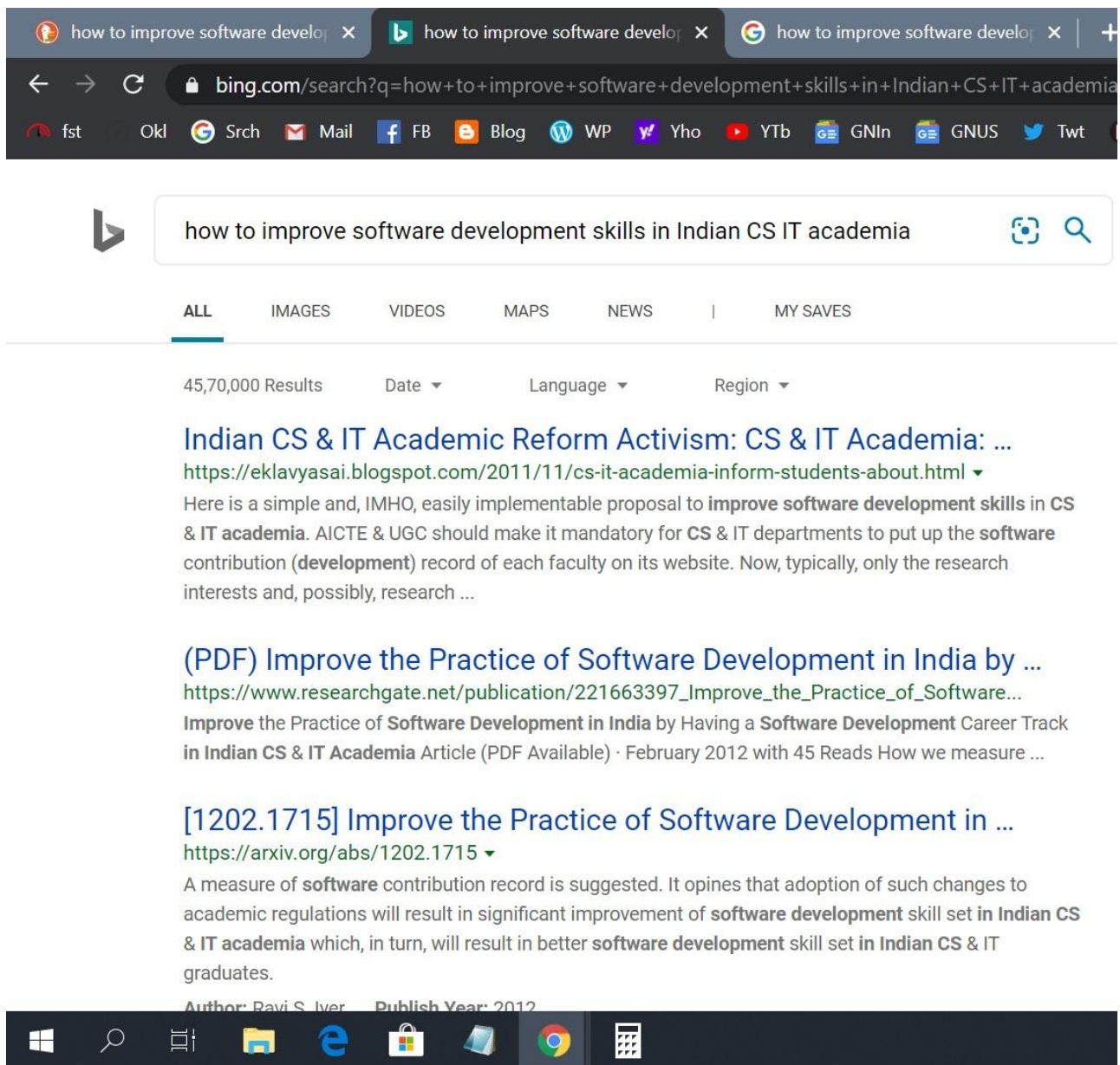
### Indian CS & IT Academic Reform Activism: Table of Contents

[eklavayasai.blogspot.com](http://eklavayasai.blogspot.com) › table-of-contents ▼  
Improving the Practice of Software Development in Indian CS & IT Academia · Top ... Poor software development skills Indian CS graduates - Google Search ...

### Ask other people a question

▶

Postina publiciv. [Learn more](#)



The full screenshots and some more information can be seen in my blog post: Internet search on: how to improve software development skills in Indian CS IT academia, gives my Dec. 2012 paper within top three results even now in Jan 2020, <https://eklavyasai.blogspot.com/2020/01/internet-search-on-how-to-improve.html> [short link: <http://bit.ly/2u1cjW1> ], 31st Jan. 2020.

Now my Dec. 2012 paper mentioned above is a preprint paper which has not been published in a peer-reviewed academic publication. However, I have truthfully recorded the rejection comments from a noted international Computer Science education academic publication, which I received when I had submitted it to them for review (2nd round, if I recall correctly). I have also given my responses to the rejection comments.

At that time, if I recall correctly, I felt that I should not modify my paper to suit what I considered to be narrow academic viewpoints of the academic reviewers. I thought that would result in significant loss in the key suggestions and thoughts I had espoused in the paper. So I simply froze the paper in Dec. 2012 as a preprint in arxiv.org and did not make any further submissions of the paper to any academic publication.

What I had not anticipated was that an earlier version of that paper (also available on arxiv.org) would get cited by one published paper (see Ref-1) and that the Dec. 2012 version (frozen/final version) would get cited by a PhD thesis (see Ref-2).

Ref-1. Ahmed Saleem Abbas et. al, "A literature review and classification of selected software engineering researches", International Journal of Engineering and Technology (IJET) – Volume 2 No. 7, July, 2012, Indian CS & IT Academic Reform (Past) Activism Blog Book

[https://www.researchgate.net/profile/Wilson\\_Jeberson/publication/265886949\\_A\\_Literature\\_Review\\_and\\_Classification\\_of\\_Selected\\_Software\\_Engineering\\_Researches/links/560f943e08ae48337517e1a8/A-Literature-Review-and-Classification-of-Selected-Software-Engineering-Researches.pdf](https://www.researchgate.net/profile/Wilson_Jeberson/publication/265886949_A_Literature_Review_and_Classification_of_Selected_Software_Engineering_Researches/links/560f943e08ae48337517e1a8/A-Literature-Review-and-Classification-of-Selected-Software-Engineering-Researches.pdf) [Short link: <http://bit.ly/31haQa0> ].

Ref-2. Ashutosh Madhukar Kulkarni, MANAGEMENT PERSPECTIVE (ACADEMIC AND ADMINISTRATIVE ASPECT) IN CAREER PLANNING OF STUDENTS DURING 2007-2013 WITH REFERENCE TO SELECTED MCA INSTITUTES IN PUNE REGION, Ph.D. Thesis, Tilak Maharashtra Vidyapeeth, Pune (Maharashtra, India), April 2015,

<http://shodhganga.inflibnet.ac.in:8080/jspui/handle/10603/86727> [Short link: <http://bit.ly/3b6xNkM> ].

But it is still quite a surprise to me to see the top search ranking given to my paper in Google Scholar search for terms related to: improve teaching of software development in Indian CS academia.

The key search terms for which Google Scholar lists my above mentioned Dec. 2012 paper in top ten results are:

1<sup>st</sup> rank: improve teaching of software development in Indian CS academia

1<sup>st</sup> rank: improve software development skills in Indian CS academia

2<sup>nd</sup> rank: improve practice of software development in Indian Computer Science academia

3<sup>rd</sup> rank: improve teaching of software development in Indian Computer Science academia

2<sup>nd</sup> rank: improve software development skills in Indian Computer Science academia

3<sup>rd</sup> rank: improve software development skills in Indian Information Technology academia

3<sup>rd</sup> rank: improve teaching of software development in Indian Information Technology academia

3<sup>rd</sup> rank: improve practice of software development in Indian Information Technology academia

1<sup>st</sup> rank: improve teaching of software development in Indian CS higher education

1<sup>st</sup> rank: improve software development skills in Indian CS higher education

1<sup>st</sup> rank: improve practice of software development in Indian CS higher education

7<sup>th</sup> rank: improve software development skills in Indian Computer Science higher education

7<sup>th</sup> rank: improve practice software development Indian Computer Science higher education

3<sup>rd</sup> rank: improve practice of software development in CS academia

2<sup>nd</sup> rank: improve teaching of software development in CS academia

1<sup>st</sup> rank: improve software development skills in CS academia

4<sup>th</sup> rank: improve practice of software development in CS higher education

3rd rank: poor software engineering skills of Indian Computer Science graduates

2nd rank: poor software development skills of Indian Computer Science graduates

10th rank: improve software engineering skills of Indian Computer Science graduates

1st rank: improve teaching of software engineering in Indian CS academia

1st rank: improve software engineering in Indian CS academia  
Indian CS & IT Academic Reform (Past) Activism Blog Book

3rd rank: improve software engineering in Indian Computer Science academia

4th rank: improve software engineering education in Indian Computer Science academia

1st rank: improve software engineering in Indian CS higher education

Given below are a few cropped screenshots showing some of these Google Scholar search results.

The screenshot shows a Google Scholar search results page. The search query is "improve teaching of software development in Indian CS academia". The page displays several search results, including:

- Articles** About 22,500 results (0.03 sec)
- Any time** filters: Since 2020, Since 2019, Since 2016, Custom range...
- Sort by relevance** and **Sort by date** options.
- include patents
- include citations
- Create alert
- Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia**  
RS Iyer - arXiv preprint arXiv:1202.1715, 2012 - arxiv.org  
... of such changes to **academic** regulations will result in significant **improvement** of **software** ... al., "Improving software practice through education: Challenges and future trends", pp.12-28, Future of ... experience of **Indian CS academia** I had some suggestions to **improve** the practice ...  
☆ 📄 Cited by 1 Related articles All 4 versions 🔄
- Incorporation and marginalization in the academy: From border toward center for faculty of color?**  
CS Turner - Journal of Black Studies, 2003 - journals.sagepub.com  
... Contributions of a diverse faculty **enhance teaching** and learning as well as contribute to the **development** ... will be addressed in the sciences, there is a need to **increase** the number ... New ways of thinking about **teaching** and research have provided spaces for women scholars to ...  
☆ 📄 Cited by 174 Related articles All 5 versions
- Software engineering education in India: Issues and challenges**  
K Garg, V Varma - ... 21st Conference on **Software Engineering** ..., 2008 - ieeexplore.ieee.org  
... addressing it will facilitate achievement of multiple goals of the industry that include **improved** quality, sustained ... Most of the **Indian** universities are **teaching** universities ... **development** services and upward movement in the value chain while sustaining and **improving** the growth ...  
☆ 📄 Cited by 28 Related articles All 5 versions
- People issues relating to software engineering education and training in India**  
K Garg, V Varma - Proceedings of the 1st **India software engineering** ..., 2008 - dl.acm.org  
... whatever learning they have and do not think about ways to **improve** the efficiency of ... pedagogy should be chosen as per the learning goals and should **increase** the effectiveness of ... There are

improve software development s x +

scholar.google.com/scholar?hl=en&as\_sdt=0%2C5&q=improve+software+development+skills+in+Ind

fst Okl Srch Mail FB Blog WP Yho Ytb GNIn GNUS Twt N

Google Scholar improve software development skills in Indian CS academia

Articles About 21,300 results (0.14 sec)

Any time  
 Since 2020  
 Since 2019  
 Since 2016  
 Custom range...

Sort by relevance  
 Sort by date

include patents  
 include citations

Create alert

**Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia**  
[RS Iyer](#) - arXiv preprint arXiv:1202.1715, 2012 - arxiv.org  
 ... of such changes to **academic** regulations will result in significant **improvement** of **software** ... al., "Improving software practice through education: Challenges and future trends", pp.12-28, Future of ... in an **Indian** educational institution, a deemed university, to **improve** the practice of ...  
 ☆ [Cited by 1](#) [Related articles](#) [All 4 versions](#) [»](#)

**Software engineering education in India: Issues and challenges**  
[K Garg](#), [V Varma](#) - 2008 21st Conference on **Software** ..., 2008 - ieeexplore.ieee.org  
 ... it will facilitate achievement of multiple goals of the industry that include **improved** quality, sustained ... in the next decade and that will seriously bottleneck in sustaining and **improving** the growth ... Second, it is not necessary that students opt for a **software development** project ...  
 ☆ [Cited by 28](#) [Related articles](#) [All 5 versions](#)

**[PDF] The role of academic and non-academic factors in improving college retention**  
[VA Lotkowski](#), [SB Robbins](#), [RJ Noeth](#) - ACT policy report, 2004 - Citeseer  
 ... differences and the factors that affect them are a step toward **improvement** in college ... various successful strategies employed at four-year postsecondary institutions to **improve** college retention ... scores can be directed into special programs that focus on **improving** their **academic** ...  
 ☆ [Cited by 923](#) [Related articles](#) [All 15 versions](#)

**People issues relating to software engineering education and training in India**  
[K Garg](#), [V Varma](#) - Proceedings of the 1st **India software** engineering ..., 2008 - dl.acm.org  
 ... employing whatever learning they have and do not think about ways to **improve** the efficiency ... in section 6. It is important to understand that subjects like **Software Engineering** cannot ... There

improve teaching of software de... x +

scholar.google.com/scholar?hl=en&as\_sdt=0%2C5&q=improve+teaching+of+software+development

fst Okl Srch Mail FB Blog WP Yho YTb GNIn GNUS Twt

Google Scholar improve teaching of software development in Indian CS higher education

Articles About 36,100 results (0.15 sec)

Any time  
 Since 2020  
 Since 2019  
 Since 2016  
 Custom range...

Sort by relevance  
 Sort by date

include patents  
 include citations

Create alert

**Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia**  
[RS Iyer](#) - arXiv preprint arXiv:1202.1715, 2012 - arxiv.org  
 ... of such changes to academic regulations will result in significant **improvement** of software ... al., "Improving software practice through education: Challenges and future trends", pp.12-28, Future of ... in an **Indian educational** institution, a deemed university, to **improve** the practice of ...  
 ☆ Cited by 1 Related articles All 4 versions

**Software engineering education in India: Issues and challenges**  
[K Garg, V Varma](#) - ... on **Software Engineering Education** and ..., 2008 - ieeexplore.ieee.org  
 ... addressing it will facilitate achievement of multiple goals of the industry that include **improved** quality, sustained ... Most of the **Indian** universities are **teaching** universities ... **development** services and upward movement in the value chain while sustaining and **improving** the growth ...  
 ☆ Cited by 28 Related articles All 5 versions

[HTML] **A decade of agile methodologies: Towards explaining agile software development**  
[T Dingsøyr, S Nerur, VG Balijepally, NB Moe](#) - 2012 - Elsevier  
 ... manage the process of **software development**", Adolph and Kruchten **develop** a grounded ... report analysis of the application of lean approaches in agile **software development**", Wang, Conboy ... types of lean applications—from practices for continuous process **improvement** to flow ...  
 ☆ Cited by 905 Related articles All 6 versions

**People issues relating to software engineering education and training in India**  
[K Garg, V Varma](#) - Proceedings of the 1st **India software** engineering ..., 2008 - dl.acm.org  
 ... by employing whatever **learning** they have and do not think about ways to **improve** the efficiency ... These facilities should include well trained **teaching** staff, access to SE tools and study ... There

improve software development s x +

scholar.google.com/scholar?hl=en&as\_sdt=0%2C5&q=improve+software+development+skills+in+I

fst Okl Srch Mail FB Blog WP Yho Ytb GNIn GNUS Twt

Google Scholar improve software development skills in Indian CS higher education

Articles About 41,900 results (0.16 sec)

Any time  
 Since 2020  
 Since 2019  
 Since 2016  
 Custom range...

Sort by relevance  
 Sort by date

include patents  
 include citations

Create alert

**Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia**  
 RS Iyer - arXiv preprint arXiv:1202.1715, 2012 - arxiv.org  
 ... of such changes to academic regulations will result in significant **improvement** of **software** ... al., "Improving **software** practice through **education**: Challenges and future trends", pp.12-28, Future of ... in an **Indian educational** institution, a deemed university, to **improve** the practice of ...  
 ☆ Cited by 1 Related articles All 4 versions

**Getting the numbers right: International engineering education in the United States, China, and India**  
 G Gereffi, V Wadhwa, B Rissing... - ... of Engineering **Education**, 2008 - Wiley Online Library  
 ... which have resulted in the adoption of a series of recom- mendations to **improve** and expand the ... Today, **India** exports US\$20 billion worth of **software** and BPO services, and this figure is ... on the low end of their re- spective global value chains, but the **high-tech** component of FDI ...  
 ☆ Cited by 187 Related articles All 17 versions

**Software engineering education in India: Issues and challenges**  
 K Garg, V Varma - ... on **Software Engineering Education** and ..., 2008 - ieeexplore.ieee.org  
 ... facilitate achievement of multiple goals of the industry that include **improved** quality, sustained ... and upward movement in the value chain while sustaining and **improving** the growth ... in/Nasscom/templates/NormalPage.aspx?id=4969 [7]. Shaw M., **Software Engineering Education** ...  
 ☆ Cited by 28 Related articles All 5 versions

**A global and competition-based model for fostering technical and soft skills in software engineering education**  
 O Gotel, V Kulkarni, M Say, C Scharff... - ... **Education** and ..., 2009 - ieeexplore.ieee.org  
 The quality of documentation improved in all cases. Summarize what went well on the project



poor software engineering skills

scholar.google.com/scholar?hl=en&as\_sdt=0%2C5&q=poor+software+engineering+skills+of+Indian

Google Scholar

poor software engineering skills of Indian Computer Science graduates

Articles About 34,300 results (0.15 sec)

Any time  
 Since 2020  
 Since 2019  
 Since 2016  
 Custom range...

Sort by relevance  
 Sort by date

include patents  
 include citations

Create alert

**Software engineering education in India: Issues and challenges**  
 K Garg, V Varma - ... 21st Conference on **Software Engineering** ..., 2008 - ieeexplore.ieee.org  
 ... This lacuna is a direct resultant of **poor Software Engineering (SE)** education and training infrastructure in the ... When given a choice of teaching a **programming** course or a **Software Engineering** course, most ... to be uninteresting to teach and are well aware of the **bad** reputation ...  
 ☆ 77 Cited by 28 Related articles All 5 versions

**People issues relating to software engineering education and training in India**  
 K Garg, V Varma - Proceedings of the 1st **India software engineering** ..., 2008 - dl.acm.org  
 ... issue in IT sector which is directly resulting from **poor software engineering** education and ... They should be well versed and experienced with **software development** and should keep ... In proceedings of International Conference on 12th **Software Engineering** Education & training ...  
 ☆ 77 Cited by 13 Related articles All 7 versions

**Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia**  
 RS Iyer - arXiv preprint arXiv:1202.1715, 2012 - arxiv.org  
 ... It is easy to see that the **poor** quality of **Indian CS&IT graduates** being prepared for **software development** jobs in **India** upon **graduation** is an artifact of **poor** quality **software engineering** education (or a complete lack of it) ...  
 ☆ 77 Cited by 1 Related articles All 4 versions

**Investigating the skill gap between graduating students and industry expectations**  
 A Radermacher, G Walia, D Knudson - ... on **software engineering**, 2014 - dl.acm.org  
 ... tended to have issues (ie, per-sonal **skills**, **software** tools, **computer science** concepts, and **software engineering** principles), and ... **Software** testing was also frequently reported ... not comment

improve teaching of software eng x +

scholar.google.com/scholar?hl=en&as\_sdt=0%2C5&q=improve+teaching+of+software+engineering+in

fst Okl Srch Mail FB Blog WP Yho YTb GNIn GNUS Twt NP

Google Scholar improve teaching of software engineering in Indian CS academia

Articles About 20,500 results (0.24 sec)

Any time  
 Since 2020  
 Since 2019  
 Since 2016  
 Custom range...

Sort by relevance  
 Sort by date

include patents  
 include citations

Create alert

**Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia**  
[RS Iyer](#) - arXiv preprint arXiv:1202.1715, 2012 - arxiv.org  
 ... of such changes to **academic** regulations will result in significant **improvement** of **software** ... al., "**Improving software** practice through education: Challenges and future trends", pp.12-28, Future of ... experience of **Indian CS academia** I had some suggestions to **improve** the practice ...  
 ☆ Cited by 1 Related articles All 4 versions

**Making computer science minority-friendly**  
[R Varma](#) - Communications of the ACM, 2006 - dl.acm.org  
 ... Several students raised issues that are important in **improving** minority students' **academic** performance and learning experiences ... The third question is how to **improve** the faculty- student relationship and **enhance** learning experiences of minority students ...  
 ☆ Cited by 64 Related articles All 6 versions

**Incorporation and marginalization in the academy: From border toward center for faculty of color?**  
[CS Turner](#) - Journal of Black Studies, 2003 - journals.sagepub.com  
 ... Contributions of a diverse faculty **enhance teaching** and learning as well as contribute to the **development** ... will be addressed in the sciences, there is a need to **increase** the number ... New ways of thinking about **teaching** and research have provided spaces for women scholars to ...  
 ☆ Cited by 174 Related articles All 5 versions

**Software engineering education in India: Issues and challenges**  
[K Garg, V Varma](#) - ... 21st Conference on **Software Engineering** ..., 2008 - ieeexplore.ieee.org  
 ... addressing it will facilitate achievement of multiple goals of the industry that include **improved** quality, sustained ... Most of the **Indian** universities are **teaching** universities ... **development** services



Any time  
 Since 2020  
 Since 2019  
 Since 2016  
 Custom range...

Sort by relevance  
 Sort by date

Include patents  
 Include citations

Create alert

**A global and competition-based model for fostering technical and soft skills in software engineering education**

O Goyal, V Kulkarni, M Say, C Scharff... - ... *Software Engineering* ... 2009 - [ieeexplore.ieee.org](#)  
 ... The quality of documentation **improved** in all cases ... Summarize what went well on the project and what didn't; determine how to **improve** the model going forward ... Conf. on Product Focused **Software Process Improvement (PROFES'07)**, LNCS 4589:20-34, Riga, Latvia, 2007 ...  
 ☆ Cited by 27 Related articles All 10 versions

**Investigating the skill gap between graduating students and industry expectations**

A Radermacher, G Walla, D Knudson - ... on *software engineering*, 2014 - [dl.acm.org](#)  
 ... students may not measure up the expectations of in- dustry companies and in **improving** the curriculum ... how IS professionals rated their current competency in several ar- eas of process **improvement** (such as ... into precisely what aspects of these categories **need improve- ment** ...  
 ☆ Cited by 78 Related articles All 5 versions

**Working across borders: Overcoming culturally-based technology challenges in student global software development**

O Goyal, V Kulkarni, C Scharff... - ... on *Software Engineering* ... 2008 - [ieeexplore.ieee.org](#)  
 ... to help students bond during the course of the project, to **improve** collaboration and ... Project Management Wiki To contain all documents and **software** artifacts; To **increase** milestone visibility and ... switch in emphasis to use a wiki and provide wiki training **improved** the following ...  
 ☆ Cited by 35 Related articles All 5 versions

**Introducing global supply chains into software engineering education**

O Goyal, V Kulkarni, LC Naak, C Scharff... - ... on *Software Engineering* ... 2007 - Springer  
 ... even after the end of their classes, still wanted to **improve** the **software** ... Richardson, I., Milewski, AE, Mullick, N., Keil, P., Distributed **Development**. An Education Perspective on the ... In: Proceedings of the 28th International Conference on **Software Engineering (ICSE'08)**, Shanghai ...  
 ☆ Cited by 31 Related articles All 12 versions

**Software Engineering Education From Indian Perspective**

R Mahanti, PK Mahanti - ... Conference on *Software Engineering* ... 2005 - [ieeexplore.ieee.org](#)  
 ... There has been a noteworthy **increase** in the number of higher education institutes ... and sufficiency of knowledge rating, the following ideas would help **improve** the quality ... These **improvement** measures if incorporated in the Universities and institutes will produce students with in ...  
 ☆ Cited by 15 Related articles All 7 versions

**The Indian software services industry**

A Arora, VS Arunachalam, J Asundi, B Fernandes - *Research policy*, 2001 - Elsevier  
 ... Though this situation has **improved** in a number of ways since 1998, there is no doubt that ... The tight labour market conditions are reflected in the 20% **increase** in wages and in attrition ... In addition to **improving** retention rates, an important part of the solution is a more efficient use ...  
 ☆ Cited by 717 Related articles All 16 versions

**The state of software engineering education and training**

N Mead, D Carter, M Lutz - *IEEE Software*, 1997 - [computer.org](#)  
 ... organizations began to incorporate modules on effective **software development** practices to **improve software** system productivity ... CMU/SEI-98-SR-011) and "Subject Matter of Process **Improvement**: A Topic of Reference Source for **Software Engineering** Education and ...  
 ☆ Cited by 18 Related articles All 11 versions

**Human resource development policy in the context of software exports: case evidence from Costa Rica**

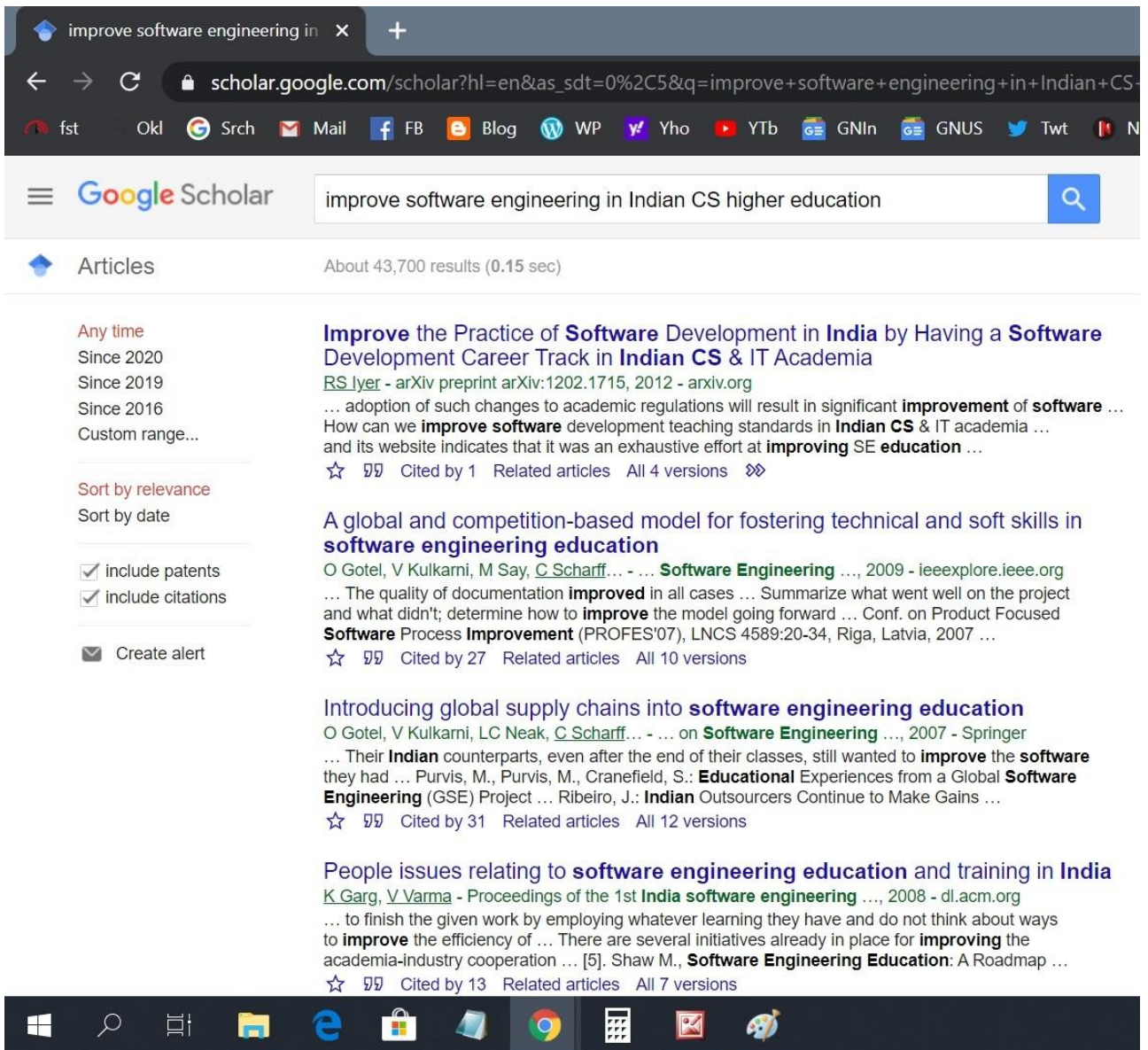
B Nicholson, S Sahay - *Progress in Development Studies*, 2008 - [journals.sagepub.com](#)  
 ... to **increase** their **computer science graduates** ... Kambhampati (2002) describes how the 1997 IT Taskforce, 'Operation Knowledge', recommended **improved** IT education in schools, and the ... have to also make the research more applied and work more closely with the **software** firms ...  
 ☆ Cited by 13 Related articles All 10 versions

**Bangalore: India's silicon city**

J Stremiau - *Monthly Lab. Rev.*, 1996 - [HeinOnline](#)  
 ... By the mid-1990s, however, **improved** telecommunica- tions technologies and **Indian** economic reforms made it ... Center for Canada enabled researchers at a Bangalore institute to **improve** a pro- cess ... The Center hopes the new process will **increase** the demand for bamboo mats ...  
 ☆ Cited by 19 Related articles All 4 versions

**Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia**

RS Iyer - *arXiv preprint arXiv:1202.1715*, 2012 - [arxiv.org](#)  
 ... of such changes to academic regulations will result in significant **improvement** of **software** ... al., **"Improving software** practice through education: Challenges and future trends", pp.12-28, Future ...  
 ☆ Cited by 13 Related articles All 10 versions



The full screenshots and some more information can be seen in my blog post: Google Scholar search too on: improve teaching of software development in Indian CS academia, and similar terms gives my Dec. 2012 paper within top few results, <http://eklavyasai.blogspot.com/2020/02/google-scholar-search-too-on-improve.html> [Short link: <http://bit.ly/2RPQodh> ], 1st Feb. 2020 (last updated 5<sup>th</sup> Feb. 2020).

[Contents of my Dec. 2012 paper PDF: <http://arxiv.org/pdf/1202.1715> are given below. In the references section broken links as of around 2<sup>nd</sup> March 2020 are indicated with terms like “[2nd Mar 2020 update: broken link]”, and at times, after the broken link, an archived link is provided or another link which seems to have the same article/content.]

## **Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia**

Ravi S. Iyer  
Software Consultant, Puttaparthi, Andhra Pradesh, India  
ravi@raviiyer.org

December 21st, 2012

### **Abstract**

*Many, but not all, Indian CS & IT academics tend to have a focus on theory and research. They do not give much importance to the practice of software development. This paper proposes an additional software development career track for Indian CS & IT academics different from the existing research oriented career track. A measure of software contribution record is suggested. It opines that adoption of such changes to academic regulations will result in significant improvement of software development skill set in Indian CS & IT academia which, in turn, will result in better software development skill set in Indian CS & IT graduates.*

*Note: The review remarks for this article by a noted international academic publication focused on CS education and the response of the author are provided in Appendix A.*

### **Introduction**

*The author is a Physics Graduate (and Physics Masters drop-out) from India who was industry-trained and later self-taught in software development. He worked in the international software industry (US, Europe, Japan, South Korea, India etc.) developing systems as well as applications software (CS & IT) for over 18 years after which he retired from commercial work. He later, mainly as an "honorary faculty/visiting faculty", offered free service of teaching programming courses (lab. courses) and being a "technical consultant" for student projects in a Maths & Computer Science department of a deemed university in India for 9 years. This paper is mainly based on this experience of the author.*

The rather odd reality of the vast number of CS & IT departments of universities & colleges in India is that the majority of the teachers in these departments focus on theory and research publications but do not give much importance to practical areas like Software Design and Programming or Coding. Therefore the practice of software development is quite poor in most Indian CS & IT departments. The sections below give references to support these statements.

### **Peer Reviewed Academic Literature Sources on Poor Software Development Skill Set in Indian CS & IT Academia**

Mahanti et al., 2005, state that in India, "Software engineering does not yet have an independent curriculum with enough durable, codified content to justify a separate undergraduate curriculum." [20]. They further state that in India, "Limited exposure to industry problems, inability to adapt course curricula to dynamic industry requirements, limited exposure to latest tools & techniques, inability to enter into emerging areas, rigid and outdated course curricula, (repetition omitted), poor industry linkages, little real-life case studies, little scope for creative learning are some of the drawbacks in the software education system in the universities."

Garg et al., 2008, conducted a survey of major software services companies in India and reported that the training programs of these companies include retraining on programming and Software Engineering (SE) as Indian academia is not able to impart these skills to the level that they expect [21]. They further state that they studied the publicly available syllabi related to SE for a large number of Indian universities and found that the focus is on theoretical aspects and "Practical aspects, best practices, recent developments are not included and students rarely get a chance for application of the knowledge and skills they learned."

## General Public Views on Poor Software Development Skill Set in Indian CS & IT Academia

The author believes that Indian CS & IT academics should also listen to views of students and others like the news media and teacher blogs on this matter. While these views may be contested as not having been validated by means of an academic/scientific peer review process the author is of the opinion that the almost unanimous voice of the general Indian public must be given some importance. The Indian public naturally expects that Indian CS & IT academics will prepare Indian CS & IT students to contribute mainly as competent software developers to the explosively growing software systems that pervade many aspects of modern life in India and the world.

Most of the students graduating out of Indian CS & IT academia end up having somewhat strong theoretical and, at times, research-oriented skills but being poor in Software Design and Programming [1, 12, 13]. The problem lies not with the CS & IT academics but with the Indian CS & IT academic system which provides career growth mainly for research output and largely ignores software contribution output [1].

A study of over 50,000 engineers who graduated in 2011 in India, very alarmingly states, "The percentage of ready-to-deploy engineers for IT jobs is dismally low at 2.68%" [19]. It further, alarmingly again, states, "An economy with a large percent of unemployable qualified candidates is not only inefficient, but socially dangerous." A Bangalore, India industry organization is planning to set up a task force to have an interface between IT industry and academia to restructure academic courses to ensure that IT graduates have skills desired by industry [18].

For this rather odd situation, where most Indian CS & IT academics/teachers do not have a strong software development skill set, to improve, Indian academic regulations should provide career advancement incentive for CS & IT academics who have a software contribution record [4]. Further, students should be informed of the software contribution record of faculty of CS & IT departments by making it mandatory for CS & IT departments to put up such information on its web site [5].

Teaching excellence in Indian CS & IT academia does not seem to be given much importance. It may be due to an obsession with research as there seems to be no significant rewards or recognition for teaching excellence whereas research excellence gives recognition, even fame at times, and career advancement [6].

How can we improve software development teaching standards in Indian CS & IT academia? The sections below examine Indian academic regulations with this objective.

## A Short Introduction to The Indian CS & IT Academic System

The University Grants Commission (UGC) is the apex academic body of India [10]. "The UGC has the unique distinction of being the only grant-giving agency in the country which has been vested with two responsibilities: that of providing funds and that of coordination, determination and maintenance of standards in institutions of higher education." [10a]. The All India Council of Technical Education (AICTE) [10d] is a professional council which operates under UGC umbrella. The UGC website states about AICTE, "The council is authorized to take all steps that are considered appropriate for ensuring coordinated and integrated development of technical education and for maintenance of standards." [10b].

"The UGC serves as a vital link between the Union and State Governments and the institutions of higher learning." [10c]. The UGC regulations/norms for appointment of academics [7] has a significant influence on the career of Indian academics of all academic streams including CS & IT who are employed in any higher education institution regulated by UGC or its professional councils like AICTE. But the extent of influence may vary depending on whether the educational institution receives government aid (funds) or not. The entry level position for a regular teacher post (as against a Teaching Assistant post) is the Assistant Professor position. The other teacher positions are Associate Professor and Professor.

In India the degrees awarded for software education by UGC/AICTE regulated academia are mainly termed as Computer Science (CS) or Information Technology (IT) degrees with an additional variant of Computer Applications degrees [15]. The Software Engineering (SE) degree/program is not well known in Indian academia. Most universities offer Software Engineering as a course along with other courses in their CS (and IT) curriculum [21].

The elite Indian Institute of Technology (IIT) institutions are independent of UGC & AICTE but they cater to only a small percentage of technical students in India. There are also thriving private software education/training institutes with a nation-wide presence sustained over decades but their certifications are different from the CS & IT degrees offered by UGC/AICTE recognized universities. This paper limits itself to UGC/AICTE recognized CS & IT educational institutions.

## UGC Appointment & Promotion Regulations for Music & Dance Discipline

Music, including the vocal art of singing, & Dance are performing arts. The teacher of these arts must be a capable performer first and should also have adequate theoretical knowledge.

This aspect of Music & Dance being a practice-oriented discipline is reflected in UGC [10] regulations for appointment of Assistant Professor, Associate Professor & Professor for Music & Dance discipline on Pages 7 - 9 of its regulations for appointment of teachers [7]. The author presumes that the regulations for promotion for Music & Dance discipline teachers will be on similar lines. These UGC regulations for Music & Dance discipline can be summarized as follows:

For the Assistant Professor post, the candidate should conform to standards similar to regular disciplines like Physics and Mathematics which are: Master's degree with 55 % Marks + NET/SLET/SET (National Eligibility Test, State Eligibility Test etc.) qualification; PhD, adequate research publication record etc. come into play for higher posts of Associate Professor and Professor.

OR

For the Assistant Professor post, the candidate should have studied under noted traditional masters, be a high grade artist of AIR/TV (Radio/Television) and have adequate theoretical knowledge; years of performance, participation in national/international seminars/workshops etc. come into play for higher posts of Associate Professor and Professor.

Specifically, practical expertise of the performer is recognized and formal academic degree qualification in Music or Dance as well as research publications are not necessary.

## Suggestion of Two Tracks for CS & IT Academics: Research Oriented and Software Development Oriented

The software development discipline is a very practice oriented discipline. Design & programming (coding) are vital skills. Of course, theoretical background is important but theory not backed by competent design & programming skill will make a software development practitioner/professional as incompetent as a musician or dancer who knows theory well but is not competent in performing music or dance.

Research is also vital for the software field. It is research that creates fundamental advances in the Computer Science & Information Technology (CS & IT) fields. Without research, the great and revolutionary force of the Internet would not have been created. This single example, itself, of the benefit of research shows how critical it is for progress in the software field. There are many, many other areas of CS & IT research that are of great importance to the software field.

In the author's opinion, we need both types of CS & IT teachers - practice oriented software development teachers & research oriented teachers. A very few teachers may excel at both, software development as well as research. But that will, in all probability, be a numerically insignificant minority among the huge number of CS & IT academics in the country.

As of now, UGC appointment & promotion regulations do not differentiate between CS & IT disciplines and disciplines like Physics & Mathematics [7]. The author could not find an equivalent regulations document for AICTE on its website but the general impression is that AICTE follows regulations similar to UGC in this regard. There is no incentive for practice-oriented software development teachers resulting in the majority of CS & IT academics being theory and research-oriented with not-so-strong software development skill set/knowledge. When the software development teacher himself is not so knowledgeable about software development the probability of students being taught software development skills well is very low.

The author suggests that UGC & AICTE regulations for appointment and promotion of CS & IT academics be modeled on the lines of that of Music & Dance discipline. The author would not like to get into the debate of CS as science vs. CS as art. His emphasis is on the software development part of CS & IT being a very practice oriented discipline like Music and the performing arts. Further, in a tightly regulated system like Indian academia, precedent for any suggested change makes it easier to consider the change. Since the regulations for Music and the performing arts already have a mechanism to cater to both the research oriented teachers and the practice oriented teachers, the author considers it appropriate to refer to the precedent and suggest a similar mechanism for Indian CS & IT teachers.

There should be two tracks for CS & IT academics - the current one for research oriented academics and another for practice oriented software development academics. Like the measure for competence in research for the (research oriented) CS / IT academic is the research publication record, the measure for competence of the software development CS / IT academic should be the quality and quantity of her open source software contribution record.

## Suggested Changes to Teacher Eligibility Tests (NET/SLET/SET) for CS & IT Disciplines

According to UGC regulations [7] the minimum requirements for an Assistant Professor appointment in engineering and technology discipline (which includes CS & IT) are a first class Master's degree in the appropriate branch of engineering and technology and qualifying in the teacher eligibility tests (NET/SLET/SET) [11]. AICTE seems to have watered down the requirement of Master's degree to a Bachelor's degree in engineering/technology discipline probably due to paucity of adequately qualified candidates applying for the Assistant Professor position. The teacher eligibility test is waived for candidates who have been awarded a Ph.D. degree [7].

The present teacher eligibility test (NET/SLET/SET) for CS / IT academics is a paper only test (though the syllabus includes C/C++ & SQL [11]) due to which an aspirant can become eligible to be appointed as Assistant Professor without having good practical software development skills! That may be acceptable for a research track CS / IT teacher. But it is unacceptable for a software development track CS / IT teacher. Aspirants who do not have good practical software development skills should NOT be appointed as software development track CS / IT Assistant Professors (or other grade Professors).

A new teacher eligibility test for software development track CS / IT academics should be introduced which will have a 50% weight-age practical test (on computer) involving programming and some amount of



design, and 50% weight-age on theory. This will ensure that software development track teacher-aspirants will have to be reasonably good in both theory and practice aspects of CS / IT.

## Allow Movement from Research Track to Software Development Track & Vice-Versa

A CS / IT academic should be able to switch track from research oriented to software development oriented if her software contribution record is appropriate. Similarly a software development oriented CS / IT academic should be able to switch track to research oriented if his research publication record is appropriate. Some CS / IT academics may have a respectable research publication record as well as a respectable software contribution record which would be a wonderfully balanced contribution record.

## How Do We Measure a Software Contribution?

This will have to be evolved over time. Software industry bodies in India like NASSCOM & CSI [8] (other countries would have other such bodies) can arrive at norms for evaluating an academic software contribution which can be updated at appropriate intervals to reflect the rapidly changing software practice. The author suggests the following for measuring (and sharing) the academic software contribution:

1. It should be open source allowing any person to download the software and use it, examine it or modify it.
2. Industry professionals should "peer review" the candidate academic software contribution using norms provided by industry bodies like NASSCOM or CSI and decide whether it is of requisite quality & quantity to be considered as a "peer reviewed" academic software contribution. Note that the contribution can be a single author contribution or a multiple author contribution like academic publications can be single author or multiple author.
3. Over time, an impact factor similar to one used by scientific journals [9] can be evolved for a "peer reviewed" software contribution. Extent of usage of software can be considered for this impact factor like citations are considered in arriving at a scientific journal's impact factor.
4. To make it difficult for contributor-aspirants to fake, plagiarize or wrongly influence peer review of software contributions, any "peer reviewed" software contribution should be open to challenge by suitable industry professionals or academics. As the software will be downloadable including its source, a challenger will be in a position to study the contribution in depth and challenge its acceptance as a "peer reviewed" contribution. The challenge can be decided by an industry body like NASSCOM or CSI appointed referee.
5. All these "peer reviewed" open source academic software contributions should be properly listed and organized in a web based repository which is openly accessible.

Involving the software industry in this "peer review" of academic software contributions may go a long way in reducing the huge academia-industry disconnect in the software field today.

## Concern of Research Rigour Being Watered Down

One concern may be that the research rigour of CS & IT departments will get diluted by having practice oriented software development track teachers. Well, we need a balance. CS & IT departments should have the "right" balance of research oriented teachers & software development oriented teachers. The "right" balance for a research-intensive department could be 80 % research oriented teachers and 20 % software development oriented teachers. In contrast, the "right" balance for a teaching-intensive department could be more like 50 % software development oriented teachers and 50 % research oriented teachers. Please note that the software development oriented teacher has to be good at theory too and has to prove his theoretical knowledge by clearing the software development track NET/SLET/SET exam.

## Allow Industry-Trained & Self-Taught Professionals to Become CS & IT Teachers By Clearing Teacher Eligibility Tests (NET/SLET/SET)

The software industry has a huge number of industry-trained and self-taught professionals who do not have a CS / IT academic qualification. Some of the biggest icons of the software industry who are world-famous like Bill Gates (Microsoft), Late Steve Jobs (Apple) and Mark Zuckerberg (Facebook) are/were self-taught. India, in particular, has a vast number of industry-trained software professionals who come from various disciplines in engineering, science, management, commerce & even arts. UGC & AICTE must recognize this reality of the software/CS & IT fields and allow interested industry-trained and self-taught professionals with significant number of years of experience in the software industry to become regular (paid) CS & IT software development academics. For such software industry professionals the requirement of a Master's degree in CS / IT should be waived like it is waived for the performing artist track in Music & Dance discipline. But the self-taught software professional MUST prove his/her capability by passing the software development track eligibility test (NET/SLET/SET) which will test both his/her theoretical knowledge as well as practical competence.

Please note that UGC regulations (and AICTE regulations too, it is presumed) allow for an "outstanding professional" of a field to be appointed as a Professor. The above mentioned suggestion is for those who are not eminent but are knowledgeable & competent industry-trained and/or self-taught software professionals.

### Industry Professionals as Visiting Faculty/Industry Consultants

Industry professionals who are not NET/SLET/SET qualified nor possess a PhD but are offering free/honorary teaching service may be accommodated as visiting faculty/industry consultants if their knowledge and skill-set are found competent by university/college & department administrative authorities. Such industry professionals who offer their services to a university/college regularly may be an insignificant minority of the CS & IT teachers of the country. They may be treated as exception cases.

### A Brief Look at Software Engineering Education, Certification and Professional Licensure in USA and Some Other Countries

The author has direct exposure to only Indian software education academia and so has focused on it for most of this article. However, it was felt that mention of software education practices in some other countries would give a larger, international perspective. So he did a small literature survey to study efforts made to ensure good software development practice in software education in USA and some other countries, and extended it to cover certification and professional licensure. The study focuses more on Software Engineering (SE) degree programs than Computer Science (CS) degree programs. Judging what aspects of this small study report could be useful in Indian environment may ideally need somebody who has direct exposure to software education field in both India and other countries like USA. This author leaves those aspects for others to consider, if they find it worthy of consideration.

SE is an established program in USA academia distinct from a CS program [22]. The SE2004 volume gives guidelines for a SE curriculum and its website indicates that it was an exhaustive effort at improving SE education quality in the USA, UK, Australia, Canada, etc. [23]. Accreditations of SE (and other engineering & technology) programs are conducted by organizations specializing in accrediting technical education. Lethbridge et al., 2007, give details of SE programs and their accreditation in USA, Canada and UK [22].

At a USA institute, "software engineering is a five year program, with students graduating with the equivalent of almost a full year of work experience." and there is collaboration between various companies and the institute on projects as part of the SE education program [25]. Its website states that its senior projects involve a team of 4 to 5 students working on challenging, real-world software issues for companies & organizations and results in a functional software tool ready to be used by the organization [26].

Stroustrup and others have adopted a "software curriculum" in a CS program with an aim to produce 'software professionals (for some definition of "professional")' and reported largely positive results from it [3].

The IEEE Computer Society offers certification of graduates as well as self-taught software development professionals by conducting certification exams [24]. It's "Certified Software Development Associate (CSDA)" certification/credential "is intended for graduating software engineers and entry-level software professionals and serves to bridge the gap between your educational experience and real-world work requirements" [24a]. Its more advanced "Certified Software Development Professional (CSDP)" certification/credential "is intended for mid-career software development professionals that want to confirm their proficiency of standard software development practices and advance in their careers" [24b]. IEEE Computer Society claims that its certification programs are "industry standard measurements of fundamental software engineering practices" and so are different from vendor-specific & product-specific certifications [24c].

Land et al., 2012, argue that current circumstances are favorable for formal certification in software engineering to be considered and state that there is growing support for IEEE CSDA and CSDP in both industry and academia [27]. They further state that these certifications are based on the IEEE Computer Society's Guide to the Software Engineering Body of Knowledge (SWEBOK) [28] which is followed by the CS and SE programs of some colleges and universities.

Laplante, 2012, mentions that 10 states of USA may soon be requiring licensure for software engineers working on systems related to "public health, safety, and welfare" [27a]. He further mentions that the professional licensure requirements for software engineers will be similar to those of other engineering professions in the states of the USA and that most components of such licensure requirements/exams for software engineers are already in place with a final component expected to be available in April 2013. However, Miller, 2012, suggests that the enthusiasm for professional licensure of software engineers be tempered with caution [27b]. He states, "Questions about professionalism and licensing in IT have a complex, international history."

Mead, 2009, gives a timeline of SE education in USA and some other countries [29], notably:

- 1980's seeing the first conference on SE education;
- 1990's seeing first class graduating with Master of Software Engineering (MSE) degree of Carnegie Mellon University, undergraduate SE programs in other universities and its accreditation, growth of industry-university collaborations, a joint committee of ACM & IEEE Computer Society being formed to promote SE as a profession, licensing being introduced by US state of Texas and a lot of controversy being generated over licensing that "continues to this day", distance learning enabling global SE education;
- 2000's seeing IEEE Computer Society adopting SWEBOK and offering CSDP certification, many universities offering international SE programs and SE education track being introduced in other conferences besides CSEET.

## Critical Views on Software Education in USA and Some Other Countries

The author felt it appropriate to share some critical views on software education in USA and some other countries from academic and general public sources.

Stroustrup, 2010, has argued that "fundamental changes to computer science education are required to better address the needs of industry", and shows the disconnect between CS academia and industry [2]. Parnas, a veteran SE academic, in an ACM Fellow profile interview in 1999, states, "Most students who are studying computer science really want to study software engineering but they don't have that choice. There are very few programs that are designed as engineering programs but specialize in software." [30]. He also states that the term software engineering is often confused with project management techniques.

Mark Tarver, who taught in UK CS academia prior to 2000, is harshly critical of programming skills of UK final year project CS graduate students who confessed to not being able to do any programming. He is also harshly critical of UK CS education in general [14].

A student, 2010, captured the feelings of the student community when he wrote, "I'm graduating with a Computer Science degree but I don't feel like I know how to program" and tried to seek advice from a professional programmer forum [16]. A USA employer/interviewer, 2011, who has hired dozens of C/C++ programmers, stated, "A surprisingly large fraction of applicants, even those with masters' degrees and PhDs in computer science, fail during interviews when asked to carry out basic programming tasks" [17].

## Conclusion

If the practice oriented software development career track, as suggested in this paper, is introduced in UGC & AICTE regulations for appointment and promotion of Indian CS & IT academics then, over time, we will have a healthy mix of both research oriented as well as software development oriented Indian CS & IT academics. We may even have significant number of software development experts from the software industry moving to Indian CS & IT academia. What a boon that will be for boosting the software development skill set of Indian CS & IT academia! It will also dramatically reduce the huge academia-industry gap that plagues the Indian software field today.

These changes, in turn, will, at least for the teaching-intensive Indian CS & IT departments, result in graduates & post-graduates of CS / IT having a good balance of theory and practice of software development with some appreciation for the research angle of CS / IT as well. Some of these graduates/post-graduates may choose to pursue research by doing a PhD in CS / IT. Some may become CS / IT academics who will be more knowledgeable about practical software development than is the case now. The majority of them will typically take up industry software development jobs for which they will be far better equipped with the required software development practice skill set than they are now.

## Acknowledgements

The author's software industry and CS doctoral student friends have provided valuable contributions to the author's Indian CS & IT academic reform activism blog: <http://eklavyasai.blogspot.in/p/table-of-contents.html>. These interactions greatly encouraged the author to attempt this rather daunting task of making a case for a software development career track in Indian CS & IT academia to improve the practice of software development in India. However, the author would like to clarify that the views in this paper are his individual views. The author thanks the reviewers of a noted academic publication focused on CS education for their critical comments which led the author to limit most of his views to the Indian context as that is what the author has studied and experienced, strengthen the article with more peer reviewed academic references and broaden its view with a brief study of software education in USA and some other countries.

## References

1. Ravi S. Iyer, (Indian) CS & IT Academia: Serious Systemic Problems?, Blog post, Sept. 2011, <http://eklavyasai.blogspot.com/2011/09/cs-it-academia-serious-systemic.html>.
2. Bjarne Stroustrup, What Should We Teach New Software Developers? Why?, "Communications of the ACM", Jan. 2010, <http://cacm.acm.org/magazines/2010/1/55760-what-should-we-teach-new-software-developers-why/fulltext>. [3rd Mar. 2020 update: The full text of article in above link has become premium content. The same article can be viewed here: [https://www.researchgate.net/publication/242499230\\_What\\_Should\\_We\\_Teach\\_New\\_Software\\_Developers\\_Why](https://www.researchgate.net/publication/242499230_What_Should_We_Teach_New_Software_Developers_Why) (short link: <http://bit.ly/what-teach> ).]
3. Bjarne Stroustrup, Programming in an undergraduate CS curriculum, WCCCE '09, May 2009, Vancouver, British Columbia, Canada, [2nd Mar 2020 update: broken link]

<http://www2.research.att.com/~bs/software.pdf> [archived link:  
<https://web.archive.org/web/20120123123212/http://www2.research.att.com/~bs/software.pdf> (short link:  
<http://bit.ly/2whPxtE> )].

4. Ravi S. Iyer, (Indian) CS & IT Academia: How To Relieve Suffering of Students, Parents and Employers, Blog post, Nov. 2011, <http://eklavyasai.blogspot.com/2011/11/cs-it-academia-suffering-students.html>.

5. Ravi S. Iyer, (Indian) CS & IT Academia: Inform Students About Software Contribution Record of Faculty, Blog post, Nov. 2011, <http://eklavyasai.blogspot.com/2011/11/cs-it-academia-inform-students-about.html>.

6. Ravi S. Iyer, (Indian) CS & IT Academia: Is Teaching Excellence Important?, Blog post, Jan. 2012, <http://eklavyasai.blogspot.com/2012/01/cs-it-academia-is-teaching-excellence.html>.

7. UGC (University Grants Commission) regulations on minimum qualifications for appointment of teachers and other academic staff in universities and colleges and measures for the maintenance of standards in higher education, 2010, [2nd Mar 2020 update: broken link]  
[http://www.ugc.ac.in/policy/revised\\_finalugcregulationfinal10.pdf](http://www.ugc.ac.in/policy/revised_finalugcregulationfinal10.pdf) [archived link:  
[https://web.archive.org/web/20111120222535/http://www.ugc.ac.in/policy/revised\\_finalugcregulationfinal10.pdf](https://web.archive.org/web/20111120222535/http://www.ugc.ac.in/policy/revised_finalugcregulationfinal10.pdf) (short link: <http://bit.ly/39INmxu> )].

8. NASSCOM: The National Association of Software and Services Companies, the premier software industry trade organization of India, [2nd Mar 2020 update: broken link] <http://www.nasscom.org/>. CSI: Computer Society of India, a leading Indian CS & IT professionals body, <http://www.csi-india.org>.

9. Impact Factor of Scientific journals: A measure which is considered by Indian academic regulatory bodies to be reflective of relative importance of a journal within its field, [http://en.wikipedia.org/wiki/Impact\\_factor](http://en.wikipedia.org/wiki/Impact_factor).

10. UGC: University Grants Commission, the apex academic body of India, <http://www.ugc.ac.in/>; 10a. University Grants Commission Mandate, [2nd Mar 2020 update: broken link]  
<http://www.ugc.ac.in/about/mandate.html> [archived link:  
<https://web.archive.org/web/20120209185301/http://www.ugc.ac.in/about/mandate.html> (short link: <http://bit.ly/39rm136> )]; 10b. Professional Councils of UGC, AICTE entry: [2nd Mar 2020 update: broken link] <http://www.ugc.ac.in/inside/pcouncil.html#AICTE> [archived link:  
<https://web.archive.org/web/20120210065710/http://www.ugc.ac.in/inside/pcouncil.html> (short link: <http://bit.ly/2TxoOkz> )]; 10c. Higher Education in India at a Glance, Feb. 2012, [2nd Mar 2020 update: broken link] <http://www.ugc.ac.in/inside/statistics.html> ; 10d. AICTE: All India Council of Technical Education: <http://www.aicte-india.org/>.

11. UGC National Eligibility Test (NET) has only one subject for CS & IT which is Computer Science and Applications: [2nd Mar 2020 update: broken link] <http://www.ugc.ac.in/inside/syllabus.html> [archived link:  
<https://web.archive.org/web/20120206105336/http://www.ugc.ac.in/inside/syllabus.html> (short link: <http://bit.ly/38gPJGI> )]. The syllabus for Computer Science and Applications subject is available here: [2nd Mar 2020 update: broken link] <http://www.ugc.ac.in/inside/syllabuspdf/87.pdf> [archived link:  
<https://web.archive.org/web/20111026213431/http://www.ugc.ac.in/inside/syllabuspdf/87.pdf> (short link: <http://bit.ly/39iTqwU> )].

The eligibility tests for Science subjects are conducted by a different organization, Council for Scientific and Industrial Research (CSIR), and that list, [2nd Mar 2020 update: broken link]  
[http://csirhrdg.res.in/Syllabi\\_NET.htm](http://csirhrdg.res.in/Syllabi_NET.htm) [archived link:  
[https://web.archive.org/web/20111230045425/http://csirhrdg.res.in/Syllabi\\_NET.htm](https://web.archive.org/web/20111230045425/http://csirhrdg.res.in/Syllabi_NET.htm) (short link: <http://bit.ly/2x3XHGV> )], does not have any CS & IT subject(s). But "Common Elementary Computer Science" questions are part of the syllabus of all Science subjects.

12. Pallab De, The State of Engineering in India, techie-buzz.com, May 2011, <http://techie-buzz.com/discussions/engineering-colleges-students-india.html>.
13. Geeta Anand, India Graduates Millions, but Too Few Are Fit to Hire, The Wall Street Journal, April 2011, <http://online.wsj.com/article/SB10001424052748703515504576142092863219826.html>.
14. Mark Tarver, Why I am not a Professor, Lambda Associates, 2007, [2nd Mar 2020 update: broken link] <http://www.lambdassociates.org/blog/decline.htm> [archived link: <https://web.archive.org/web/20111014155153/http://www.lambdassociates.org/blog/decline.htm> (short link: <http://bit.ly/38RxBEr> )].
15. Computer Science (CS) / Computer Science & Engineering (CSE) and Information Technology (IT) departments in India offer:
- B.E./B.Tech.(CS/CSE), M.E./M.Tech.(CS/CSE)
  - B.E./B.Tech.(IT), M.E./M.Tech.(IT)
  - B.C.A. and M.C.A. degrees.
- [B.E. - Bachelor of Engineering, M.E. - Master of Engineering B.Tech. - Bachelor of Technology, M.Tech. - Master of Technology  
B.C.A. - Bachelor in Computer Applications, M.C.A. - Master in Computer Applications.]*

Model syllabus for:

- B.E./B.Tech. in CSE: <http://www.aicte-india.org/downloads/mugcomputersc.pdf>
  - MCA: <http://www.aicte-india.org/downloads/mcadegree.pdf>
16. A CS student, I'm graduating with a Computer Science degree but I don't feel like I know how to program, programmers.stackexchange.com, May 2010, <http://programmers.stackexchange.com/questions/43528/im-graduating-with-a-computer-science-degree-but-i-dont-feel-like-i-know-how-t>.
17. Dan Kegel, How To Get Hired -- What CS Students Need to Know, www.kegel.com, June 2011, <http://www.kegel.com/academy/getting-hired.html>.
18. V.S. Karnic, Tackling the 'graduate but unemployable' syndrome in India, IANS India, Jan. 2012, [2nd Mar 2020 update: broken link] <http://in.news.yahoo.com/tackling-graduate-unemployable-syndrome-india-063206080.html> [this seems to be same article: [http://www.thaindian.com/newsportal/business/tackling-the-graduate-but-unemployable-syndrome-in-india\\_100593013.html](http://www.thaindian.com/newsportal/business/tackling-the-graduate-but-unemployable-syndrome-in-india_100593013.html)].
19. Varun Aggarwal, Aspiring Minds' National Employability Report – Engineering Graduates, 2011, [http://www.aspiringminds.in/docs/national\\_employability\\_report\\_engineers\\_2011.pdf](http://www.aspiringminds.in/docs/national_employability_report_engineers_2011.pdf).
20. Rupa Mahanti, P. K. Mahanti, "Software Engineering Education From Indian Perspective," cseet, pp.111-117, 18th Conference on Software Engineering Education & Training (CSEET'05), 2005, [2nd Mar 2020 update: broken link] <http://www.computer.org/portal/web/csdl/doi/10.1109/CSEET.2005.30> [this seems to be same paper: <https://doi.org/10.1109/CSEET.2005.30> ].
21. Kirti Garg, Vasudeva Varma, "Software Engineering Education in India: Issues and Challenges," cseet, pp.110-117, 21st Conference on Software Engineering Education and Training, 2008, <http://dx.doi.org/10.1109/CSEET.2008.36>.
22. Lethbridge, T.C. et. al., "Improving software practice through education: Challenges and future trends", pp.12-28, Future of Software Engineering, FOSE '07, 2007, <http://dx.doi.org/10.1109/FOSE.2007.13>.

23. Software Engineering 2004 (SE2004), Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, <http://sites.computer.org/ccse/>.
24. IEEE Computer Society Certification & Training, [2nd Mar 2020 update: broken link] <http://www.computer.org/portal/web/certification/home> [archived link: <https://web.archive.org/web/20121204071121/http://www.computer.org/portal/web/certification/home> (short link: <http://bit.ly/2IeFrME> )]; 24a. IEEE Computer Society Certified Software Development Associate (CSDA) credential, <http://www.computer.org/portal/web/certification/csda>; 24b. IEEE Computer Society Certified Software Development Professional (CSDP) credential, <http://www.computer.org/portal/web/certification/csdp>; 24c. IEEE Computer Society certification FAQ and Resources, [2nd Mar 2020 update: broken link] <http://www.computer.org/portal/web/certification/resources> [archived link: <https://web.archive.org/web/20120905162456/http://www.computer.org/portal/web/certification/resources> (short link: <http://bit.ly/2PFjcDV> )].
25. Reichlmayr, T.J., "Collaborating with industry: strategies for an undergraduate software engineering program", pp.13-16, Proceedings of the 2006 international workshop on Summit on software engineering education SSEE '06, 2006, <http://dx.doi.org/10.1145/1137842.1137848>.
26. Rochester Institute of Technology, Senior Project done by every RIT Software Engineering senior, [2nd Mar 2020 update: broken link] <http://www.se.rit.edu/senior-project> [archived link: <https://web.archive.org/web/20121024130849/http://www.se.rit.edu/senior-project> (short link: <http://bit.ly/2TdTrN1> )].
27. Land, S.K. et al., "Software engineering certification in today's environment ", pp. 50 - 54, IT Professional, May-June 2012, Volume: 14, Issue: 3, <http://dx.doi.org/10.1109/MITP.2012.21>; 27a. Laplante, P.A., "Professional Licensure of Software Engineers", Side-bar article in [27]; 27b. Miller, K.W. "IT Professionalism and Licensure: Yes, But...", Side-bar article in [27]
28. IEEE Computer Society's Guide to the Software Engineering Body of Knowledge (SWEBOK), [2nd Mar 2020 update: broken link] <http://www.computer.org/portal/web/swebok> [archived link: <https://web.archive.org/web/20121112184706/http://www.computer.org/portal/web/swebok> (short link: <http://bit.ly/2VBaZ7v> )].
29. Mead N. R., "Software engineering education: How far we've come and how far we have to go", pp. 571-575, Journal of Systems and Software, Volume 82, Issue 4, April 2009, <http://dx.doi.org/10.1016/j.jss.2008.12.038>.
30. ACM Fellow Profile: David Lorge Parnas, May 1999, [2nd Mar 2020 update: broken link] <http://www.sigsoft.org/SEN/parnas.html> [archived link: <https://web.archive.org/web/20121008191209/http://www.sigsoft.org/SEN/parnas.html> (short link: <http://bit.ly/2vvxtfA> )].
- 

## Appendix A

### Review Remarks of Noted International Academic Publication and Response of Author

After a lot of consideration the author decided to take the unusual step of sharing review remarks for this article by a noted international academic publication focused on CS education. The editor-in-chief of the publication graciously provided permission to share the valuable review remarks of the knowledgeable reviewers in this appendix. The author thanks him and the publication for this kind gesture.

As the author sees it, the stake holders of software education imparted by UGC/AICTE recognised institutions in India are:

1. Students (& Parents): They invest their time and pay the tuition fees.
2. Teachers/Academics: They are supposed to be knowledgeable and do the primary task of imparting appropriate knowledge to students.
3. Employers (Industry): They use the products of the education system students-turned-graduates) to contribute to economic work and provide a livelihood for these students-turned-graduates.
4. Funding agencies & regulators, namely MHRD (Ministry of Human Resource Development, <http://mhrd.gov.in/>), UGC & AICTE: They provide the tax payer contributed money for higher education (e.g. as UGC grants) and try to maintain good standards of education. They also look at nationwide issues and society issues like the needs of the country and imparting ethics. Further, they try to promote an environment that will encourage good education (attract good teachers, provide job security to teachers, give students a safe environment, etc.)

The author is of the opinion that this article/paper may not be easily accepted in a forum primarily controlled by one of the stakeholders here, namely the teachers/academics, as it is somewhat critical of them even though the criticism is mainly directed at the system rather than the individual academics. But the article/paper may find a lot of acceptance in forums of some of the other stakeholders especially students, parents, industry and perhaps even MHRD.

This article went through 2 rounds of review with the noted international academic publication. The first round feedback was incorporated in this version of the article, which is what was submitted for the 2nd round review (except for a minor difference in the title of the article). It was not found suitable for the publication in the 2nd round review though the reviewers had some appreciation for the article.

The author views the 2nd round reviewer comments and his response to them as a debate between CS academic viewpoint and industry software developer viewpoint. (The author is not an academic but a software design & development practitioner from the industry who helped out a CS department in an Indian educational institution, a deemed university, to improve the practice of software development in it). He opines that some stakeholders of software education like students, parents and industry practitioners may want to read this debate and that it may contribute to a better understanding of this issue among the public at large. Very importantly, the reviewers provided very knowledgeable comments which contribute significantly to the discourse on the topic. Therefore the author decided to share the review remarks and his response to them in this appendix.

Reviewer: 1

Comments to the Author

Author: Firstly, thank you very much for your valuable remarks. They have contributed significantly to my understanding of this issue from an international perspective.

Reviewer1: This paper makes an interesting and controversial case for creating career tracks in the Indian CS&IT academia for faculty who, instead of pursuing theoretical research, would opt for establishing academic credentials based on their output of open source software. Much of this seems to reflect from the author's personal journey from being a non-CS graduate, with extensive experience in the CS&IT Software Development industry, and then trying to "fit in" into the academic make-up of an Indian University which seems to be heavily regulated centrally.

Author: I disagree with a part of the last statement. I consider myself to be an accomplished industry-trained and self-taught software industry technical consultant, who, mainly as an Honorary Faculty/Visiting Faculty, provided free teaching and guidance service to students in software development/engineering and thereby contributed to strengthening the practice of software development/engineering in a CS department in India. I was not and am not interested to "fit in" the Indian CS & IT Academic Reform (Past) Activism Blog Book



academic make-up of any Indian university at all - my intention was to help students learn the practice of software development well, and I believe I succeeded in no small measure in that regard. From my experience of Indian CS academia I had some suggestions to improve the practice of software design & development/engineering in it which I put forward in this article/paper.

Reviewer1: Many issues here: the first and foremost, whether the specific situation in Indian academia is worth publishing in ---publication-name-description-blinded--- with a much broader international audience. On the other hand, given that such situations exist in several countries, this is a good discussion starter to bring a broader awareness to the issues faced and the possible (in my mind, skewed) solution being proposed.

From what I gather, the main problem seems to be in the area of software engineering education.

Author: Yes, but the whole gamut of software engineering involving design, development, testing etc. and not just a software development process/life-cycle theory course. A published research paper mentioned that in India, "Most universities offer SE as just one of the courses along with other Computer Science courses.", Kirti Garg, Vasudeva Varma, "Software Engineering Education in India: Issues and Challenges," cseet, pp.110-117, 21st Conference on Software Engineering Education and Training, 2008, <http://dx.doi.org/10.1109/CSEET.2008.36>.

Reviewer1: While, on one hand, most CS&IT faculty in India seem to be primarily devoted to theoretical research, there seems to be a practice among academia of relaxing the qualifications for hiring CS&IT faculty due to the "paucity" of qualified faculty. It is easy to see that the poor quality of Indian CS&IT graduates being prepared for software development jobs in India upon graduation is an artifact of poor quality software engineering education (or a complete lack of it).

Author: I tend to agree with these views in the Indian context.

Reviewer1: The solution being proposed, to create UGC regulated mandates for separate faculty career tracks, is rather controversial, and perhaps misplaced when one sees it in the larger context of the role of academia. Much of the problem should, and can, be addressed by creating a well qualified pool of faculty in the CS&IT disciplines, software engineering included. Additionally, along with the faculty, to create educational tracks in CS&IT Departments at universities to teach software engineering curricula. Without a presence of these two things, it seems like a proposal to create accommodations for industry professionals to enter non-research tracks in academia, is misplaced and a poor solution.

Author: It certainly is *\*not\** a proposal focused on creating accommodations for industry professionals to enter non-research tracks in academia. It is a proposal to provide career growth incentive for Indian CS & IT academics to excel in the practice of software development instead of focusing on research publication output and ignoring excellence in practice of software development. Industry professionals being accommodated in a non-research software development career track is a secondary and optional part of what this paper proposes, which in my opinion, has significant value for improving the practice of software development in Indian CS & IT academia.

Reviewer1: This can be detrimental to the intrinsic health and make-up of an entire higher educational system. The case is made, based on faculty in the Performing Arts which tends to be one of the few outliers in this regard. Even in Performing Arts, there is much resistance. Ordinary, day-to-day practitioners of the art seldom attain faculty status even in the Performing Arts.

Author: In India, it is common to see faculty of the performing arts deliver a performance to the public which leaves students, parents and the public in general in no doubt as to the practical skill of the performing arts faculty. In marked contrast, there is huge amount of doubt in the mind of students, parents and the public in general about the practical software development/engineering skill of most Indian CS & IT academics.

Reviewer1: Besides, there are existing models that accommodate "both" classes of faculty in a single framework that are present outside India that need to be examined. For example, giving academic credit for software artifacts during the promotion and tenure process is widely promoted by the Guidelines published by the Computing Research Association (in the USA). The author should take a look at that.

Author: It is interesting and it will be good if Indian academic regulations take note of it and provide significant academic credit for software artifacts. I agree very much with the view expressed therein that, "Assessing artifacts requires evaluation from knowledgeable peers." In the Indian context, in my opinion, at least in the short term, it is the software industry which has the capacity to provide enough numbers of knowledgeable peers to evaluate software artifacts produced by Indian CS & IT academia.

Reviewer1: Much of the surveys presented in the paper are about the state (or lack there of) of software engineering education in India. To suggest that a government regulated body create a "practitioner track directed to serving a dimension of a transient and evolving industry" and further put into place specific software-based evaluation metrics for the hiring, promotion, and career advancement of such faculty is a bizarre idea that makes for an excellent blog post, or an opinion piece.

Author: I humbly submit that as a practitioner of software development I find it bizarre that, in the key regulations that govern Indian CS & IT academia, there is zero career growth incentive to excel in the practice of software development. That, in my opinion, is the key reason for such poor quality of software engineering/software development skills in most Indian CS & IT academics.

Further, I believe that the huge growth of the software design & development field in India is enough reason to seriously consider the suggestion of a specific Indian academic career track which focuses on software design & development. It may be a very strange suggestion for academia in general but the software revolution in the past few decades has changed India and the world quite a bit and Indian academia may need to look at new ways to effectively handle its duties of teaching Indian students the vital skill of software design & development.

Reviewer1: This submission to ---publication-name-blinded---, in this sense, is misplaced.

---

Reviewer: 2

Comments to the Author

Author: Firstly, thank you very much for your valuable remarks. They have contributed significantly to my understanding of this issue from an international perspective.

Reviewer2: It is good to see a paper addressing issues relating to the relevance and quality of computing education at a national level in India, and the challenges in preparing competent practitioners for the local IT industry. Such a discussion has potential to be of interest to --- publication-name-blinded--- readers. However the solutions proposed fail to take into account several critical issues.

The move beyond computing as CS, EE or IS to a broader set of cognate computing disciplines in a wider family as noted in the ACM 2005 overview report. Therefore institutions need the flexibility to adapt curricula to meet both local conditions and international standards. A highly rigid national framework militates against such adaptability, and thus we see private organisations filling the gaps by providing vocationally focussed certifications. If the core degree learning provides a sound underpinning education, then maybe this is ok? Shackelford, R., Cassel, L., Cross, J., Davies, G., Impagliazzo, J., Kamali, R., Lawson, E., LeBlanc, R., McGettrick, A., Slona, R., Topi, H. and vanVeen, M. Computing Curricula 2005 The Overview Report including The Guide to Undergraduate Degree Programs in Computing, Joint Task Force ACM, AIS, IEEE-CS, New York, 2005, 46.

Author: The above document seems to be a very well thought out and well researched document from a North-American perspective. However I do not know how well it can help solve the problem of very poor practice of software design & development in Indian CS & IT academia which is heavily influenced by the regulations and guidelines of UGC & AICTE.

Reviewer2: It needs to be recognised that the nature of the CS/SE divide is historical and long standing, [as is the role of programming in CS] but the divide is arguably one of the strengths of CS and SE that both theory and practice must interrelate in the achieving of outcomes – so education should recognise this in some form. Cf. for instance the discussion below: Lister, R., Berglund, A., Clear, T., Bergin, J., Garvin-Doxas, K., Hanks, B., Hitchner, L., Reilly, A. L., Sanders, K., Schulte, C. and Whalley, J. Research Perspectives on the Objects-Early Debate. SIGCSE Bulletin, 38, 4 (Dec 2006), 146-165.

Author: Noted.

Reviewer2: “These distinctions can be traced back to the origins of the discipline, and early schisms “between the logicians and the technicians”, (Clark, 2003) depending upon whether one came from a more theoretically oriented mathematical background, or a more practically oriented engineering background”.

The role of the SE discipline has always been problematic, and the tensions between the body of knowledge, the evolving nature of practice in the field, what skills should be taught and the nature of the academy and its value systems has always been an issue. But University systems are by their nature international, and local solutions which focus primarily on the vocational teaching mission [with a goal of producing immediately productive ‘drones for industry’] without doing equal justice to the research mission, are likely to result in reputational damage to the institution. Cf. the discussion below:

Clear, T. Software Engineering and The Academy: Uncomfortable Bedfellows? SIGCSE Bulletin, 36, 2 (June 2004), 14-15.

Author: I read some other messages in the above article:

"Reflecting upon how this experience had enriched his teaching upon his return to the academy, he also noted that few engineering educators possessed any experience of engineering practice."

...

"If we consider medicine as an analogous profession, have not the medical educators themselves completed clinical practice requirements? Would doctors who had never practiced be regarded as credible professors of clinical medicine? Why do we privilege the doctoral qualification over the practice credentials in the case of our software engineering professors?"

...

"Being prepared to recruit ex-practitioners without PhD qualifications, and recognise their value in non-traditional ways may be strategies vital to success in teaching a quality software engineering programme."

In my opinion the above views match the views expressed in my paper.

Reviewer2: But I doubt that measuring an academic’s software capability and contribution by open source software production is practicable. (Although for the ‘R’ statistical software package, newly contributed and specialised statistical modules are formally peer reviewed before acceptance). For instance Open Source is but one mode of software development. What of the skills of developing proprietary software in teams?

Author: The problem with proprietary software would be availability of source code for reviewers. I guess this would be similar to proprietary research work which is not published in academic research publications and therefore may not contribute to an academic's career growth directly.

Reviewer2: What of domains of application? What of experience of software engineering gained through research involvement with software development firms? Such partnership models are often applied as academics move away from regular software development, and perhaps through supervision of development by students, to a more theoretical, SE process, SE practice or managerial SE research focus.

Author: The software contribution record that I suggested looked only at software contributions. It can be discussed whether it should be expanded to include some of the above suggestions and the manner in which it should be included.

Reviewer2: The reality of the research teaching divide is also endemic in the academy, as developed in the paper below, and as recently observed by the Business School Accreditation Body AACSB's Blue Ribbon Committee in the report below:

Clear, T., Valuing Computer Science Education Research? [Invited Presentation]. in 6th Baltic Sea Conference on Computing Education Research (Koli Calling 2006), (Koli, Finland, 2006), Uppsala University, Uppsala.

AACSB. Discussion Paper – Relationship Between Research and Teaching, AACSB Blue Ribbon Committee on Accreditation Quality 2011, 1-5.

Author: Noted.

Reviewer2: In other systems when hiring academics, their skill sets are carefully considered against the needs of the department. It is unlikely that an academic with no knowledge of software engineering practice would be hired to teach a software engineering course, although they may teach a math course or a course in theoretical CS.

Author: Unfortunately what Clear,T. wrote (mentioned earlier), "few engineering educators possessed any experience of engineering practice" applies very well to Indian CS & IT academia. Career growth incentive to them to improve in software engineering practice may lead them to make efforts to do so. Otherwise it is natural that they will be attracted only towards producing research publications as that provides career growth.

Reviewer2: As presented, while there is active debate in many of these areas, the paper does not fully address the issues [disciplinary, institutional, cultural, political] that are obstacles to implementing such a nationwide programme of change in the Indian Higher Education System. It poses a challenging set of questions in the context of the author and his perspective on computing education in his country, and how practice and theory gap might be reduced. To that extent the issues are important to all computing educators, so a debate could certainly be had. As it stands unfortunately the paper lacks the dimensions to productively lead that debate.

With a more considered set of proposals and a wider understanding of the academic setting and how to effect change in a complex context with local and global dimensions, there could be a stronger candidate for publication.

---- end of contents of paper <http://arxiv.org/pdf/1202.1715> ----

# Ten Computer Science (CS) & Information Technology (IT) higher education policy changes to improve practice of software development in India

Associated blog post date: 25th Aug. 2014, link: <http://eklavyasai.blogspot.com/2014/08/ten-computer-science-cs-information.html> , short link: <http://bit.ly/ten-changes>

*Note that the contents of this article may slightly differ from associated blog post. Similarly, contents of later articles in this book may slightly differ from their associated blog posts. Further note that most of the blog posts and comments on the posts have author name as Eklavya Sai Maalik which I (Ravi S. Iyer) was using as a pseudonym in the past.*

I submitted the following today (25th August 2014), with the same title as that of this blog post, to an appropriate Indian govt. online forum (which is accessible only to its registered users - but any Indian citizen can easily register in it for free):

- 1) Have two career tracks for CS/IT academics: Research Oriented and Software Development Oriented
- 2) Like the measure for competence in research for the CS/IT academic is the research publication record, the measure for competence in software development of the CS/IT academic should be the quality and quantity of his/her software contribution record.
- 3) Change teacher eligibility tests (NET/SLET/SET) for CS & IT disciplines to have 50% weight-age practical test (on computer) involving programming and some amount of design, and 50% weight-age on theory.
- 4) Allow industry-trained & self-taught professionals who are not academically qualified in CS/IT to become CS & IT Teachers by clearing teacher eligibility tests (NET/SLET/SET)
- 5) Encourage industry professionals to contribute as well paid visiting faculty/industry consultants.
- 6) Study CS/IT professional licensure & certification in USA and other technologically advanced countries and explore possibility of introducing similar CS/IT licensing and certification exams in India.

For more on points 1 to 6, see my paper, "Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia", whose text contents have been given earlier in this book.

7) Teachers who create course material for a course (as against using course material from other sources) should have their course material reviewed by other teachers just like research papers are reviewed. Appropriate career growth related credit should be given to teachers whose course material gets a good review. The review may include student assignment submissions including source code, so that reviewers get an idea of how students are benefiting from the course.

8) Lab. courses should be evaluated as strictly as theory courses. It should have external examiners like theory courses. There should be no hesitation in failing students who fare poorly in assignment submissions and exams. Administrators should examine lab. course result patterns and compare it with theory course result patterns. Very high success rates in lab. courses contrasted by quite different success rates in theory courses should ring alarm bells and invite investigation.

9) Lab. course credits should be at least equal to theory course credits. Further, the number of lab. courses should be roughly equal to the number of theory courses. [I believe, the typical current ratio is 4 to 5 theory

courses and 2 lab. courses in a semester, with theory courses having 3 credits and lab. courses having 2 credits.]

10) The practice of relegating lab. courses to junior teachers must be abandoned. Given the vital importance of practice in CS/IT, senior teachers should teach lab. courses.

For more on points 7 to 10, see the next post below: Concrete Suggestions for Measuring Teaching Quality in Practice-Oriented Computer Science/Information Technology streams.

## **Concrete Suggestions for Measuring Teaching Quality in Practice-Oriented Computer Science/Information Technology streams**

Associated blog post date: 4th June 2013, last updated on 16th Sept. 2014, link:

<http://eklavyasai.blogspot.com/2013/06/concrete-suggestions-for-measuring.html> , short link: <http://bit.ly/conc-suggs>

At the outset, I request the kind indulgence of readers for blowing my own bugle a little, with the intention of substantiating my case.

Yesterday I received an email from a recently passed out M.Tech. (Computer Science) student (I don't know if exam results have been declared yet but the student is a high-flyer and will, in all probability, pass). He thanked me for my C++ classes which he felt played a significant role in him clearing a job interview and getting a job offer which he intends to take up.

--- Start passed out student message extract ----

I must thank you for getting this offer because my interview was solely based on C++ programming and very little of algorithms. Even the latter was with respect to data structures that are part of the C++ STL. Your C++ classes, and at a later stage your slides were of great help to understand the basics of C++. Therefore, I felt this urge within to inform you about this.

--- End passed out student message extract ----

I would have taught him the C++ course when he was in Ist M.Sc. (Maths) in academic year 2009-10. The department is a Mathematics & Computer Science department. Many of the two year M.Sc. (Maths) program students continue on to do the two year M.Tech. (Computer Science) program. [BTW in March 2012 I parted ways with the above-mentioned department and educational institution.]

Now I would not like to comment much on why the tech. interview (as per the student's description) did not cover algorithms in some depth. IMHO, ideally there should have been a balance of questions on both algorithms and the particular programming language skills they are interested in, which was C++ in this case.

However, the reality of the situation is that the practical programming skills imparted to the student via the lab. course was instrumental in him clearing the job interview and landing the job. So the teacher concerned would seem to have done a good job and would earn the appreciation of academic administrators.

But the way things are run in UGC/AICTE regulated educational institutions in India, with accreditation agencies like NAAC being powerful forces, good teaching quality like in the above case, are not recorded in any fashion in their evaluation metrics. If I am not mistaken, the teaching quality assessment according to

UGC/ACITE norms is supposed to be or can be a self-assessment! Naturally all teachers will be assessed as good or excellent teachers then :-) (unless there are many theory exam failures).

The key measure that is recorded is the research publication record. NAAC (the key national accreditation agency for UGC institutions, I believe) seems to be focused mainly on the research publication record of institutions it accredits and grades. So academic administrators have become obsessed with research publication record to ensure that the institutions they administer get a good NAAC rating. This results in a scenario where academic administrators put immense pressure on teachers to publish research papers, and almost ignore promoting excellence in teaching quality.

Now, in my special case, I had retired from commercial work in Sept. 2002, and was offering honorary service of teaching lab. courses in a 'deemed' university. I was utterly disinterested in an academic career or in academic research (my designation during my stint was Honorary Staff, Honorary Faculty and finally Visiting Faculty). I was only interested in my spiritual career and part of that involved serving students by teaching them what they needed to know using my knowledge base of 18 years international software industry practitioner experience and also studying new knowledge areas required to teach specific lab. courses allotted to me. So my focus was on excellence in teaching lab. courses. I have had the satisfaction of teaching them to the best of my ability, given the circumstances, and receiving the love and gratitude of the students in return. From a spiritual and ex-software industry practitioner/professional career point of view I consider it to be a very successful stint (9 years) of spiritual and industry-quality service.

But from an Indian academic system point of view my question is, will the typical Indian CS/IT academic focus on excellence on teaching lab. courses like the C++ course I taught? IMHO, the clear answer is No (there may be some exceptions, of course - I am talking about the typical Indian CS/IT academic). As his/her bosses are mainly bothered about NAAC grading and UGC/AICTE norms for which the only thing that really matters is research publication record.

Further, lab. courses are looked down upon as unimportant courses which can be handled by junior staff like Teaching Assistants or Research Assistants (Research Scholars). The evaluation of lab. courses are very lenient. Rarely is a student failed and many students get the top grade. Theory courses have paper examinations with external evaluation. That is considered far more important in the (UGC/AICTE) Indian academic system. Theory courses have more credit than lab. courses. [If UGC/ACITE publish statistics about failures in CS/IT lab. courses in the institutions they regulate, I am quite sure it will be a miniscule number. However, the failures in CS/IT theory courses will be, I am quite sure, a not insignificant number, in comparison.]

This is very strange for a very practice oriented field like Computer Science/Information Technology. As the above freshly passed out student interview example clearly shows, the vital skill that is assessed during typical CS/IT job interviews is the practical skill. Of course, theory is important too and ideally there should be a balance between theory and practice. But, IMHO, an administration policy where the teaching of the practice of software development (lab. courses) is relegated to junior teachers and considered unimportant, is a disastrous education policy for a practice-oriented field like Computer Science/Information Technology.

What can we do to rectify the situation? Teaching Quality must be measured even if the measures may have some flaws in it (like the flaws in measuring research contributions). Over time the flaws in measuring teaching quality could be addressed/controlled. I have a few concrete suggestions for measuring teaching quality of lab. courses in Indian (UGC/AICTE) CS/IT academia:

- 1) Teachers who create course material for a course (as against using course material from other sources) should have their course material reviewed by other teachers just like research papers are reviewed. Appropriate credit should be given to teachers whose course material gets a good review. The review may include student assignment submissions including source code, so that reviewers get an idea of how students are benefiting from the course.

2) Lab. courses should be evaluated as strictly as theory courses. It should have external examiners like theory courses. There should be no hesitation in failing students who fare poorly in assignment submissions and exams. Administrators should examine lab. course result patterns and compare it with theory course result patterns. Very high success rates in lab. courses contrasted by quite different success rates in theory courses should ring alarm bells and invite investigation. If the teachers concerned are found to have done an overly lenient job of evaluation they should be penalized.

Additional suggestions to improve the teaching of the practice of software development, not related to measuring teaching quality, are:

- a) Lab. course credits should be at least equal to theory course credits. Further, the number of lab. courses should be roughly equal to the number of theory courses. [I believe, the typical current ratio is 4 to 5 theory courses and 2 lab. courses in a semester, with theory courses having 3 credits and lab. courses having 2 credits.]
  - b) The practice of relegating lab. courses to junior teachers must be abandoned. Given the vital importance of practice in CS/IT, senior teachers should teach lab. courses.
- 

Initially I had suggested student feedback as one measure of teaching quality, as follows:

\*) Student feedback about the course should be collected. Industry training institutes heavily use participant feedback to get an idea of the quality of teaching and suggestions for improvement. Academia can learn from them. Yes, there may be some serious concerns like students using it as a weapon against the teacher, and the teacher targeting classes where he/she receives bad feedback. But there surely must be ways to mitigate such issues. Students are the key customers/stakeholders, usually paying customers, of the education system. Teachers in academia must learn to respect this fact and learn to accept criticism from students about their teaching, just like teachers in industry training institutes respect (or are forced to respect) criticism from participants/students of their courses.

--- end suggestion ---

I received vehement criticism from a US CS academic (over two separate mails which he allowed me to share with others). I have shared below the key parts of the mail exchange:

The US CS academic wrote:

It may not have occurred to you but some students punish good teachers for being demanding and for not being soft at grading. I have seen the most outrageous \*anonymous\* student feedback for a professor who caught many cheating. The easiest way for a professor to get a good rating from anonymous feedback is to lower standards and give higher grades.

My response was: Oh Lord! While I thought that certainly there were possibilities of students using the anonymous feedback as a weapon against the teacher, I did not know that they actually do go that far. You have the experience of such feedback systems in academia - I don't (though I do have industry feedback experience, which I think is a very different ball game). I am quite shocked to know of this. Hmm.

The US CS academic wrote back:

Note that anonymous accusations traditionally are absolutely disregarded or in cases even illegal under Western law. Except, it seems for professors. Disgusting.



My response (slightly edited to give more clarity) was: Thank you so much for bringing out this aspect so clearly. I think an academic who gets targeted via such anonymous feedback/accusations would go through horrifying emotional torture. He/She would feel there has been no justice, no chance for him/her to prove his/her case. Hmm.

I really need to pause, think hard, and ensure that my blog post does reflect this horrifying abuse possibility. And then explore whether there is a "safe" way to use student-customer feedback as one measure of teaching quality. As of now, I am quite clueless about suggestions on how to prevent abuse of student feedback like the cases you mentioned and how to address/prevent/resolve the 'disgusting' nature of anonymous feedback accusations being considered as a negative measure of teaching quality.

Thanks again for the reality-bite.

--- end mail exchange extracts ---

After learning about the US academic's experience with abuse of anonymous student feedback to punish good teachers I realize that my earlier view that "there surely must be ways to mitigate such issues" may be wrong. I am not so sure any more about ways to prevent or even mitigate such issues in the Indian UGC/AICTE regulated institutions context. [For the 'cheating' example shared above one could ignore feedback from such students but they could influence their student-colleague-friends to give bad feedback.] My experience of teaching in academia is limited to an educational institution with a focus on human and spiritual values where the students were exceptionally well behaved and had tremendous respect for their teachers. Perhaps there is a very real possibility of such abuse of anonymous student feedback to punish good teachers in some other Indian educational institutions.

Direct (non-anonymous) student feedback used as a measure of teaching quality, when negative, has the possibility of seriously damaging the human relationship between student and teacher. So I don't think that can be considered.

Therefore I withdraw the suggestion of using student feedback as a measure of teaching quality.

## **Discussion on Concrete Suggestions for Measuring Teaching Quality ...**

Associated blog post date: 5th June 2013, link: <http://eklavyasai.blogspot.com/2013/06/discussion-on-concrete-suggestions-for.html> , short link: <http://bit.ly/disc-conc>

This post is a follow up discussion post to the Concrete Suggestions for Measuring Teaching Quality in Practice-Oriented Computer Science/Information Technology streams post (given earlier in this book).

A correspondent who is now doing a PhD in Computer Science in a US university wrote (slightly edited):

Interesting views from you on teaching lab courses, balance between lab and theory, and evaluating teaching techniques and teachers. If you are interested, here is how I have perceived teaching in the US. Note that this is just my view and I have taken only a few courses/classes and come to this conclusion by mere observation.

I firmly believe that teachers must be evaluated on an yearly basis. Here in the US, at the end of every semester the students go through an anonymous survey where they rate their instructors and the facilities in the lab. There is a "Best Teacher Award" every semester. There is also a "Best Researcher Award" too. In

spite of all such measures in place, there are good and poor teachers here as well. I have come to understand that being a good teacher is a matter of attitude more than anything else. I have seen the busiest of professors spend inordinate amounts of time and effort to make themselves understood and convey the course content very effectively. On the other hand, I have seen/heard of professors who take their tenure-ship for granted and just breeze through the classes. These teachers get poor ratings in surveys but nothing seems to be done about it. While I believe that we need anonymous online evaluation of instructors and facilities, I also believe that the system can only do so much and inspirational/excellent teaching is a matter of teacher-attitude.

Regarding programming courses, let me go through how its done in the US. I have been a teaching assistant (TA) for basic Java, C++ and Assembly Language programming courses. There is proportional weight given to different aspects, like Exams, Assignments, Labs, Quiz(surprise tests in class). The instructor meets the students in class, twice a week for a total of 3 - 4 hours. He/She is also available for office hours during other times of the week. The instructor and TAs are available by email anytime during the semester. The introductory programming courses have one lab every week where the TA is there to help the students understand the lab. The instructor usually designs the lab and the students get all the help they need to complete the labs. The labs are evaluated, though very leniently. Every fortnight there is an assignment. And these are tough assignments that require at least a few hours to a couple of days to complete successfully. All of these are graded very objectively and there is always a check for plagiarism. One important aspect of the courses here - the instructor decides the syllabus and the evaluation criteria. This is a bit tricky to implement in India but seems to work in the US in most cases. A fifteen week course usually ends up having 8 assignments, 14 labs, 10 quizzes and 3 exams. There is also a website for the course where assignments, labs, lecture slides, solutions, reviews for exams and course material is regularly uploaded. I mentioned all of this to tell you the different aspects that go into making a programming course successful, here in the US. I am not certain that there is this much of rigour that goes into programming courses in most Engineering institutions in India. No wonder we have very different programmers in US compared to India.

I believe technology (course website, plagiarism checks, etc.) and resources (labs, TAs) are definitely needed to make a programming course successful. But the most important aspect is the instructors' attitude towards the course. With all of this technology and resources, we still have effective and ineffective instructors and that's just a matter of attitude.

I (Ravi) responded (slightly edited):

Thanks for the detailed info. and thoughtful comments. They are very useful.

Allowing teaching excellence to be an optional goal left to the attitude of the teacher, IMHO, is a disastrous management approach to quality education. A good management approach \*must\* reward excellent teaching and punish bad teaching. Such rewards and punishments will ensure that teachers will develop the right attitude towards their primary duty of teaching else they will be either left behind in their career or even asked to pursue another career option.

In my view, the key point here is that the academic administrative authorities in your US university have some measure of teaching quality (as viewed by the student-customers of the system). It may not be a perfect measure like academic research publication record is not a perfect measure of academic research contribution. This measure along with other informal assessments allows the academic administrative authorities to reward good teaching, not reward indifferent teaching, and punish bad teaching. Whether the academic administrative authorities in your US university are doing so or not, would be a confidential matter, and so not known to us. However, by the view you have that there are some ineffective instructors (bad teachers) in the system in your university, it seems that the academic administrative authorities are not acting effectively on the student feedback or are not in a position to do so.

In UGC/AICTE Indian educational institutions, for CS/IT lab. courses, there seems to be no proper measure available to academic administrative authorities to differentiate between good, indifferent and bad lab. course teachers. The course grades (given to students) seem to be given very leniently, so they are not a proper measure. This creates a situation where there is no incentive whatsoever for good teaching or any disincentive for bad teaching for CS/IT lab. courses. Is it any wonder then that most Indian CS/IT graduates are poor in software development skills/practice of software development?

The recently passed out M.Tech. (CS) student who gave the comments mentioned in the Concrete Suggestions for Measuring Teaching Quality in Practice-Oriented Computer Science/Information Technology streams post (given earlier in this book) wrote (slightly edited):

Sir,

I went through your suggestions. I would like to propose a few other minor things which are as follows:

1) A programming lab. course should not just teach the syntax and features of a language but the endeavour should be to teach the small things, which make a big difference, such as indentations while writing code, writing easy to understand comments, choosing meaningful variable names and planning the modules before writing a piece of code. I think these programming aesthetics should be part of the evaluation process.

2) This one may not always be possible. However, with a little motivation from the teacher and some interest from the students it is not as humongous a task as it may apparently be. I am talking about including a small software development project as part of the course. This can be a group project. At the end of the course the students can be made to give a presentation about their software to external examiners. (I do not know if you remember but we had a similar component, although without external examination, in the Web Programming using ASP.NET and C# lab. course that you had conducted for us. Also, 5 of us had volunteered the same year to write an assembler for a hypothetical architecture that was used in our Systems Programming text book to explain the concepts of assemblers, linkers, loaders, etc. We wrote the bare bones of it in C and this has undergone evolution over the years in the hands of subsequent batches. Now, I believe all the features of an assembler that were discussed in the course have been implemented and the software can be used to demonstrate an assembler to students. However, I am not sure if it still has a GUI.)

I (Ravi) responded (slightly edited):

Thanks for your comments.

On point 1, I agree. But I feel this should be part of the two semester 'C' programming course that is taught at B.Sc. level, which is the first programming course lab. in the Department of Mathematics & Computer Science system (in that educational institution).

On point 2, it is useful input. I agree. The issue from a teacher's perspective is time. Sometimes there is a bare minimum of topics that must be covered in a lab. course, and that leaves no time for group projects in such lab. courses, especially in time-wise, very tight educational systems.

## **Wrote to Prime Minister, Shri Narendra Modi, on improving practice of software development in Indian CS & IT academia**

Associated blog post date: 28th May 2014, link: <http://eklavyasai.blogspot.com/2014/05/wrote-to-prime-minister-shri-narendra.html> , short link: <http://bit.ly/wrote-to-PM>

I sent the following message yesterday (27th May 2014) to Honourable Prime Minister (of India), Shri Narendra Modi, via the PM website link: <http://pmindia.nic.in/feedback.php>

Subject: Education

Comments: Sir, in an interaction you had with TV18 during your campaign, you had mentioned about employability issues in our education system. I share a similar concern in the field of Information Technology in India. I request you or your aides to go through my pre-print paper on arXiv.org, Cornell University, USA, titled, Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia. Abstract link: <http://arxiv.org/abs/1202.1715> and Paper link: <http://arxiv.org/ftp/arxiv/papers/1202/1202.1715.pdf>.

Please note that for the past many months (if not a year or two) this pre-print paper of mine is the first or second result for Google search for the following terms

poor programming skills Indian CS graduates

poor programming skills Indian CS and IT graduates

poor programming skills Indian Computer Science graduates

improve programming skills Indian CS graduates

improve programming skills Indian software graduates

--- end comments ---

A little while after I submitted the message, I received a standard email acknowledgement from PMO Public Response Unit, [do-not-reply@gov.in](mailto:do-not-reply@gov.in), mentioning, "Your Mail is being reviewed."

So far, so good. Let's see whether matters go beyond this perhaps automated response.

*[8th Feb. 2020 Update: As far as I can recall, matters did not go any further - i.e. there was no further response from PMO (Prime Minister's Office) on this matter.]*

## **Poor software development skills Indian CS graduates - Google Search Results**

Associated blog post date: 25th Jan. 2013, link: <http://eklavyasai.blogspot.com/2013/01/poor-software-development-skills-indian.html> , short link: <http://bit.ly/poor-swdev>

I felt it appropriate to share on this blog a mail I sent on December 27th, 2012 to the (Indian) Ministry of Human Resource Development, Ministry of Communication & Information Technology, UGC, AICTE and AIB-ITE heads (top academic administrators of the country) and Planning Commission Nodal officer. I also felt it appropriate to share that I received a prompt response from the office of a Minister of State from the Ministry of Communication & Information Technology, which was copied to all the other mail recipients of the original mail. The response acknowledged receipt of the mail and mentioned that the mail will be forwarded to another official in the same ministry for "his kind perusal and needful further action thereof". I have not had any further response from them so far (January 25th 2013).

I am quite happy that at least the mail was acknowledged by a Union minister of state's office. Hopefully it may result in some discussion/debate on the matter.

The mail was sent on December 27th, 2012 and its contents are given below:

Dear Gentlemen,

Teaching software development skills effectively to current and future generations of Indian Computer Science (CS) and Information Technology (IT) graduates is, in my humble opinion, vital for India's material/economic development. In fact, software has become so pervasive now that software development skills are becoming necessary for graduates of other streams like Bio-sciences/Bioinformatics, Physics, Chemistry, as well as, I presume, most, if not all, engineering streams (Electrical engineering, Civil engineering, Chemical engineering etc.)

However the current standards of teaching of software development skills in most of Indian academia are quite unsatisfactory.

Google search for

poor software development skills Indian CS graduates

Or

improve software development skills Indian CS graduates

gives my academic paper/article, "Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia", as the first result.

As you gentlemen are very influential in formulating and implementing higher education and Information Technology policy for the whole of India I felt it appropriate to draw the above Google search results top ranking of my paper for the above terms, to your kind attention. Please note that a friend of mine who lives in the USA confirmed today that Google Search from a USA computer for the above terms gives the same top ranking result for my paper.

If you would like to read my paper (latest version dated 21st December 2012) hosted on the arxiv.org e-print facility of Cornell University, USA, you may please visit the (currently) top Google search result link for search terms mentioned above or the links given below:

Abstract: <http://arxiv.org/abs/1202.1715>

PDF: <http://arxiv.org/pdf/1202.1715>

[Of course, the Google search results ranking change over time and so my paper may lose its current top ranking for the above mentioned terms. But I presume it will retain this ranking for the next few days, at least.

Also note that the short description that comes below the search results has text (date) from the older version of the article but the result link takes one to the latest version of the article.]

For your reading convenience, I have also attached to this mail a copy of my paper.

Please note that the review remarks for my article by a noted international academic publication focused on CS education and my response to the same are provided in Appendix A of the paper.

I felt it appropriate to include the introduction in the Appendix of my paper in this mail itself, below:

### **Appendix A**

#### **Review Remarks of Noted International Academic Publication and Response of Author**

After a lot of consideration the author decided to take the unusual step of sharing review remarks for this article by a noted international academic publication focused on CS education. The editor-in-chief of the publication graciously provided permission to share the valuable review remarks of the knowledgeable reviewers in this appendix. The author thanks him and the publication for this kind gesture.

As the author sees it, the stake holders of software education imparted by UGC/AICTE recognised institutions in India are:

1. Students (& Parents): They invest their time and pay the tuition fees.
2. Teachers/Academics: They are supposed to be knowledgeable and do the primary task of imparting appropriate knowledge to students.
3. Employers (Industry): They use the products of the education system (students-turned-graduates) to contribute to economic work and provide a livelihood for these students-turned-graduates.
4. Funding agencies & regulators, namely MHRD (Ministry of Human Resource Development, <http://mhrd.gov.in/>), UGC & AICTE: They provide the tax payer contributed money for higher education (e.g. as UGC grants) and try to maintain good standards of education. They also look at nationwide issues and society issues like the needs of the country and imparting ethics. Further, they try to promote an environment that will encourage good education (attract good teachers, provide job security to teachers, give students a safe environment, etc.)

The author is of the opinion that this article/paper may not be easily accepted in a forum primarily controlled by one of the stakeholders here, namely the teachers/academics, as it is somewhat critical of them even though the criticism is mainly directed at the system rather than the individual academics. But the article/paper may find a lot of acceptance in forums of some of the other stakeholders especially students, parents, industry and perhaps even MHRD.

This article went through 2 rounds of review with the noted international academic publication. The first round feedback was incorporated in this version of the article, which is what was submitted for the 2nd round review (except for a minor difference in the title of the article). It was not found suitable for the publication in the 2nd round review though the reviewers had some appreciation for the article.

The author views the 2nd round reviewer comments and his response to them as a debate between CS academic viewpoint and industry software developer viewpoint. (The author is not an academic but a software design & development practitioner from the industry who helped out a CS department in an Indian educational institution, a deemed university, to improve the practice of software development in it). He opines that some stakeholders of software education like students, parents and industry practitioners may want to read this debate and that it may contribute to a better understanding of this issue among the public at large. Very importantly, the reviewers provided very knowledgeable comments which contribute significantly to the discourse on the topic. Therefore the author decided to share the review remarks and his response to them in this appendix.

--- end extract from Appendix of paper ---

If you gentlemen have any comments on my paper, critical or otherwise, I will be glad to know of them. I would like to mention that my intention in putting out this paper (and this mail) is to help Indian CS & IT students learn software development skills well.

Thanks & Regards

Ravi S. Iyer

Software Consultant

--- remainder of email signature snipped ---

## Comments (of former CS academic from UK) on Software Development Career Track Paper

Associated blog post date: 19th Mar. 2012, last updated on 26th Jan. 2013, link: <http://eklavyasai.blogspot.com/2012/03/improve-practice-of-software.html> , short link: <http://bit.ly/impr-pract>

Given below are comments I received from Dr. Mark Tarver for an earlier version of my academic paper which had the title, "Improve the Practice of Software Development by Having a Software Development Career Track in CS & IT Academia".

March 23rd, 2012

Dr. Mark Tarver's Comments on an earlier version of the paper, Abstract: <http://arxiv.org/abs/1202.1715v2>, PDF: <http://arxiv.org/pdf/1202.1715v2>

Dr. Mark Tarver formerly of [lambdassociates.org](http://lambdassociates.org), who is also a former UK CS academic, was kind enough to share his valuable comments on the above paper. He further kindly permitted me to post his comments (and my response) suitably edited for a blog post, here.

Let me take this opportunity to thank Dr. Tarver for his very frank and very well expressed views in his articles on CS academia. These articles and some of Prof. Bjarne Stroustrup's articles and a few private mail exchanges that an internationally renowned software development practitioner and CS academic kindly had with me, gave me the courage to challenge Indian CS/IT academic mind-set regarding software development practice.

Of course, my views may have flaws. Prof. Stroustrup and Dr. Tarver may disagree with part or all of my views. But, in my considered opinion, the situation is so dire in Indian CS/IT academia, in general, that it has almost become dysfunctional! There may be some exceptions, of course, but, in general, in my considered opinion, CS/IT students are getting a very raw deal. CS/IT academia in other countries may also be partly dysfunctional. We urgently need discussion followed by concerted attempts of corrective action to improve life for CS/IT students & employers.

Dr. Mark Tarver (MT) wrote: I read your paper with interest; and I agree with the thrust of what you say.

Ravi S. Iyer (RI) responded: That is vital input for me. Thanks again.

MT: Two points to consider.

1. Industry is rarely innovative and trend-setting in terms of practice and computer-language development. Practices linger on in industry due to the costs of adoption long after they are left behind by best academic practice.

RI: I tend to agree with your views here, though I do not have wide spread exposure to world wide academia to comment on "best academic practice".

MT: Lisp, Prolog, ML were all developed at university. It was industry, not academia, that kept Cobol alive for a long time.

RI: Industry marches to the beat of a different drummer, the money making drummer :). I agree.

MT: Hence placing the criteria for academic promotion in the hands of commerce is likely to stultify innovation in software practice.

RI: Very interesting point. In India, software practice in academia is quite poor. While we do have an Open Source community in India, as far as I know, it is nowhere near the thriving Open Source community that say, the US, has. So, I felt that industry should get involved in the "peer review" in the initial years.

Over time, if Indian CS/IT academia itself acquires the capability to "peer review" software development contributions, then industry can be dropped from the "peer review" process. But, as of now, in India, industry is the only community, I think, that can deliver on software development "peer review" for the huge numbers of students that India has.

MT: I would say it needs to be put in the hands of the open source community.

RI: I think, this would be a wonderful option for countries like the US and UK. And, if it proves itself as a sustainable and successful process, then other countries like India can follow their lead.

MT: 2. There is a third class of teacher who is neither productive in terms of writing (commercially significant) code or in writing papers but who is simply an excellent teacher. This sort of person is totally under-served at the moment.

RI: I entirely agree. In fact, I have at first hand seen how teaching excellence is neglected. You may want to read my blog post, CS & IT Academia: Is Teaching Excellence Important?, <http://eklavyasai.blogspot.com/2012/01/cs-it-academia-is-teaching-excellence.html> [short link: <http://bit.ly/teach-excel> ] where I conclude by writing, "I think I am an old school guy who feels that the FIRST and FOREMOST DUTY of a TEACHER is to TEACH and TEACH WELL."

## CS & IT Academia: Serious Systemic Problems?

Associated blog post date: 13th Sept. 2011, last updated on 28th Jan. 2012, link: <http://eklavyasai.blogspot.com/2011/09/cs-it-academia-serious-systemic.html> , short link: <http://bit.ly/serious-s>

The unfortunate reality of the vast number of CS & IT departments of universities & colleges in the country is that the Professors of these departments are strong in theory and research publications but not so strong in practical areas like Software Design and Programming or Coding. Naturally the students graduating out of CS & IT academia also end up having somewhat strong theoretical and research-oriented skills but being weak in Software Design and Programming.

But the biggest employer for CS & IT graduates is the software industry which mainly needs strong software design and programming skills. Yes, there are some industry research jobs too where research orientation would be very useful but such jobs are very few.

Most parents want their children to get a good industry job after they finish their CS & IT education. They get shocked when their children even after graduating/post-graduating with a B.E./B.Tech./BCA/M.Tech./MCA do not land a job. The reality is that campus recruitment is the main mechanism through which 'freshers' get jobs. Otherwise it is very tough for a CS & IT stream graduate to get a job!



Why is that? Is there not enough demand for CS & IT skills? Well, the problem is that most of the ‘fresh’ graduates simply do not have the skills to do software programming jobs. The large software companies put the ‘freshers’ through intensive in-house training before they are deployed on ‘live’ projects.

The ‘freshers’ who fail to get recruited at campus, many times, pay through their noses and go through an additional private i.e. not government recognized, “job oriented” training course which gives them “job oriented” skill set. But should not CS & IT academia provide that? That is what parents and students expect. Even common people expect that an M.Tech. in CS/IT or an MCA fresher will be able to straight away do programming tasks.

The reason lies in policies adopted by AICTE/UGC which are the vital academic regulatory agencies of the country. The primary factor for the career growth of any academician lies in his/her research work measured in terms of research publications. The CS & IT academician naturally has to focus on research to enhance his career. If he focuses on programming then he may stagnate. And so, in general, the teachers themselves neglect programming in favour of research. When the teacher himself does not focus on programming how will the students learn programming well? The system instead encourages the students also to do research and discourages too much interest in programming!

So is there any solution? There should be policy changes which will encourage CS & IT academicians to grow in software design and programming. For such changes to come AICTE/UGC MUST have significant industry representation. As of now AICTE/UGC policy makers seem to be research oriented Professors with limited exposure to software industry’s engineering needs.

But will AICTE/UGC be willing to listen to such words? In all probability the research oriented Professors will look down upon such views. So what do we common folk, non Professors, do? Well, we should approach the Ministry of Human Resources & Development (MHRD). We common folk pay taxes and it is OUR TAX MONEY which MHRD gives to AICTE/UGC and which flows down to universities and colleges. MHRD is answerable to the TAX PAYER and is answerable to the parents and students about why most, but not all, CS & IT graduates are weak in practical design and programming skills.

MHRD should insist that AICTE/UGC policy making bodies MUST have significant software industry representation. MHRD should ensure that AICTE/UGC norms for career growth for CS & IT academicians give a growth path where software design & coding expertise is given as much importance as research publications. Perhaps AICTE/UGC rules should be changed to allow non-research-oriented software engineering professionals from industry to become university teachers.

This problem is not peculiar to India alone. U.S. academia too has significant problems. Here is the world-famous Inventor of C++, Prof. Bjarne Stroustrup’s article in the reputed “Communications of the ACM” publication on changes needed in CS education: What Should We Teach New Software Developers? Why?, <http://cacm.acm.org/magazines/2010/1/55760-what-should-we-teach-new-software-developers-why/fulltext> [short link: <http://bit.ly/teach-new> ] [9th Feb. 2020 update: The full text of the article in above link has become premium content. The same article can be viewed for free, as of now at least, here: [https://www.researchgate.net/publication/242499230\\_What\\_Should\\_We\\_Teach\\_New\\_Software\\_Developers\\_Why](https://www.researchgate.net/publication/242499230_What_Should_We_Teach_New_Software_Developers_Why) (short link: <http://bit.ly/what-teach> ).]

Prof. Bjarne Stroustrup is a Distinguished Professor & holder of College of Engineering Chair in Computer Science at Texas A&M University. He is an ACM fellow as well as an IEEE fellow. For more details, see: [http://en.wikipedia.org/wiki/Bjarne\\_Stroustrup](http://en.wikipedia.org/wiki/Bjarne_Stroustrup), <http://www.stroustrup.com/>.

---

Some comments from the post are given below.

Eklavya Sai Maalik (Ravi S. Iyer) wrote on September 17, 2011 at 12:04 PM

One comment I received today over email was:

The disconnect between industry and academia exists in almost all trades- not just on CS. It seems, this is the major shortcoming of Indian education.

--

I (Eklavya Sai) think that it is really acute in the case of CS & IT. There is a huge "job oriented" private (not government recognised) IT training institute industry in India. I don't think we have anything similar in scale for, say, Mechanical Engineering or Electrical Engineering.

I am sure most of us would know of many cases of CS/IT graduates doing private "job oriented" courses after graduation, to land a job. But it may help readers of this blog post if I illustrate the matter with a couple of cases.

a) Recently an MCA fresher from Kerala was telling me that many of his colleagues including him are struggling to get a job. Note that his MCA was a 9 semester one (4.5 years) after he finished his B.Sc.! Each semester fees was around Rs. 6,500/- I am told.

But one student did a 3 month "job oriented" course, after his MCA, paying Rs. 30,000/-. The MCA fresher I spoke to put it well saying that the 30,000/- paid was fine even if it was for 3 months only **BECAUSE THE STUDENT GOT A JOB!!**

He then told me that during interviews he is asked questions like do you have experience with PHP frameworks (one IT technology example). Unfortunately he has only done a PHP/mysql mini-project without using any framework. He then told me that while doing his MCA course, for electives they were offered courses like Speech Processing and Finite Automata (or something like that). They asked the teachers whether they could do something like PHP frameworks course - the teachers said that course is not offered as elective! Please note that this is for an M.C.A. (Master of Computer Applications) course and not an M.Tech. (Computer Science) course.

I think the reason for poor technology hands-on courses in government (AICTE/UGC) recognised CS/IT academia is the seemingly blinkered research only mind-set of AICTE/UGC policy makers. Their influence on the university teachers is MASSIVE. I am not blaming the teachers. What can they do if their top masters provide career growth only through research?

I asked the MCA fresher whether it would have been better for him to have done a "job oriented" IT course in a private (non government recognised) training institute after his B.Sc. instead of his M.C.A. He agreed that it would have been a better option from the point of view of getting a job!!! He probably would have been a 3.5 year experienced IT professional now with a significant bank balance instead of still being an IT job seeker!

b) Some time ago a B.Tech./B.E. in Computer Engg. or something like that from a Hyderabad Engg. college was struggling to get a job. He too followed the same route. i.e. He paid through his nose for a "job oriented" course in Hyderabad again and then, if I recall correctly, landed a job.

So, in conclusion, while there may be an industry-academia disconnect in most streams (trades), I feel that in CS/IT the disconnect is really glaring.

---

Eklavya Sai Maalik (Ravi S. Iyer) wrote on September 21, 2011 at 12:38 PM

A friend sent this link: <http://www.joelonsoftware.com/articles/ThePerilsofJavaSchools.html>

Interesting article!

My take is that there should be differentiation: IT schools and CS schools.

Java or PHP or ASP.Net & Rapid Appl. Dev. tools in general, IMHO, are perfect for IT work. In my industry days we would call it Application Software and at that time people used to say that 80 % of programming work/business is Application software. While developing App. software, understanding and fulfilling customer needs is the most important - efficiency of code is required but not that vital. So one needs app. domain knowledge (e.g. banking, insurance, manufacturing etc.) to some extent. Then all you want is a tool that allows you to quickly develop the app. - the simpler the tool the better - so long as it does not compromise too much on efficiency and maintenance (lifetime of software). I am now quickly learning the PHP framework CodeIgniter with the same mindset - I am picking up just what I need to know to generate a simple database form oriented web app.

Pointers, Recursion ... are VITAL for a Computer Science (CS) course. During my industry days we would call CS kind of stuff as "system software" and this was around 20% of the programming work/business. The CS/Sys. software guys are the ELITE. Their earnings are also typically far more than the application software guys. They develop the core systems or tools on top of which IT/application software guys work. So the OS and the compilers/interpreters and the internet protocols will typically be done by the CS guys. The IT guys do the business apps. on top of it.

I agree with the author that Java is not a good choice for a CS course.

However, many times students/colleges/universities have this massive problem of jobs after they finish their courses. The smart guys have the capacity to do pointers & recursion. But many of the students are not that smart. So they struggle through such courses and either get 'weeded out' or worse, as is typical in India, somehow manage to finish with a CS degree but without the CS aptitude. The latter struggle to get a job. They don't know any IT to get IT jobs (unless they get swept in during 'bulk intake' campus interviews of major IT software consultancy companies) and are not good enough in CS to get system software jobs.

I think it is these realities of jobs for the students that makes many CS schools, at least in India, have some IT courses as well.

The elite tech. schools like an IIT in India, for example, don't need to worry about such things as they get capable students to start with.

---

Eklavya Sai Maalik (Ravi S. Iyer) wrote on September 22, 2011 at 9:40 PM

A friend (Kartick Vaddadi) wrote (some words are edited):

- Paul Graham has written eloquently about this: <http://www.paulgraham.com/hp.html>

<http://www.paulgraham.com/hundred.html>

<http://www.paulgraham.com/desres.html>

You really should read these, despite the length -- they are worth every word.

- This is part of the reason I did not do a PhD - the goal of academia seems to be publishing papers, or at any rate things that sound suitably academic and formal. Type systems, for example, are a good topic for academics, because you can do any amount of intellectual gymnastics about them. So, if A is the set of all integers and B is the set of all chars, struct { int x; char y;} takes on values represented by the set A x B (cross product, the math operator).

I remember arguing with an IIT prof about this, asking: this is all fine, but how does it help me write better programs? He didn't have an answer, but he didn't care -- for him, "formal" is an end to itself. --- snip ---

(Unrelated to the above) In my last week of stay at IIT, as I was sitting in my advisor's office with the bookshelves containing all the previous years PHd and Mtech thesis impressively bound and looking like a formidable intellectual arsenal, I was depressed. Who is going to read all this? Whereas the Orkut app I worked on at Google that had 1 lakh users may not win me a Turing Award. But it lets people keep in touch with their girlfriends and boyfriends! Did all this theoretical gymnastics help a single person in any concrete way?

In other words, the motivations of CS academia are wrong. When that is the case, is there any surprise if the results are not ideal? --- snip ---

People sometimes say that industry is where you do routine work and academia is where you learn and build knowledge for the next generation of systems. Except that there's a lot of dog work in academia, too, at least to the tiny degree that I'm aware of academia. My MTEch thesis involved building a different kind of debugger for an academic language called Haskell. This language has some impressive properties, for sure, and it was good to be exposed to that way of thinking, but the debugger I built is more of routine work than any kind of invention or research or original contribution to the world.

Perhaps it would be more useful if you can take one idea from Haskell and apply it to mainstream languages. Useful in the sense that it helps people. If I build a more efficient and cheaper solar panel but nobody uses it, whom does it help?

You often get good things by combining existing ideas in a slightly different way, but academia looks down on this kind of work. The kind of work I'd like to do is to look at what people use and ask, "how can I improve this?" This most likely entails using an existing idea rather than coming up with a new one, just as a painter uses existing colors in a new way rather than producing a tube of paint in a different shade.

Academia focuses on coming up with something new, rather than something useful. If you look at the status quo and come up with an idea on how to improve it, is the idea any less useful if it wasn't original? Coming up with new ideas should be a means to an end (improve the technology that people use), but academia made it an end to itself.

I think I've finally understood -- at least partially -- what people mean by the disconnect between industry and academia. But in my view, the fault lies with academia, which has confused its means for its ends.

To summarize, academia focuses on things that are (a) formal and (b) original while forgetting what really matters, which is whether it's useful.

- Masters should be only for people who have worked for a few years. A student should first learn how things are done before taking an advanced course (Mtech) or doing research.

- A different kind of innovation from core OS is interfaces. I get irritated when I read an article saying "the iPad doesn't do anything that my Unix workstation in the early 80s didn't". Did your Unix workstation have more than a hundred million users who are repelled by the fact that traditional computers are too complex and don't work right in ways that people care about? It's easy to forget this since we're all geeks, but spending a little time with my mom is enough. I can suddenly see how badly designed things are from her perspective.

This is the most interesting thing at work. Who knows, I might even make it the focus of my career.

- Leaving aside theory for the sake of theory, I think a good education should equip people with a deep understanding of how computers work, at all levels of the stack -- why something is done in a particular way

at one level of the stack, how else it could have been done, what the tradeoffs are, etc. Most BTech/MCA people don't have this knowledge.

Nor do they care. They are more interested in getting a job. Which is fine, but I have no sympathy for these people. Didn't get a job? Tough luck, you weren't interested enough in learning computer science. And I certainly have no sympathy for CS depts that don't give students the kind of deep knowledge that I'm talking about.

- I understand that the MCA student you mentioned in the comment was not well served by the MCA. Agreed. But why couldn't the student answer, "did you use PHP?" with "No, but I used Ruby on Rails, and CherryPy, and Closure, and a similar Java framework, and one or two templating languages, so I can pick up PHP before the day is out"?

To me the fault is as much with the student as the department. Anybody can point fingers and say, "but they didn't teach me!". That's not useful. Anybody who's interested in learning will find a way to learn, rather than blaming someone else, even if the other party is at fault.

Phew! :)

---

Eklavya Sai Maalik (Ravi S. Iyer) wrote on September 22, 2011 at 9:43 PM

I (Eklavya Sai) replied:

Thank you for your valuable views.

Will read the Paul Graham articles later.

You wrote, "This is part of the reason I did not do a PhD - the goal of academia seems to be publishing papers, or at any rate things that sound suitably academic and formal."

That is how it comes across, isn't it? Actually the goal of academia should be to teach!!! And there are many academicians, I think, who focus on that. But for academic career growth publishing record is the metric. I mean, I have heard academicians being referred to as 'he has published 400 papers', which is supposed to mean that he is a great academic.

I think this is a systemic issue for academia - especially Indian academia which is governed at the top-level by AICTE/UGC/Ministry of Human Resources & Development. They put out the 'norms' for recruitment and promotion of Indian academicians. For example, see [archived link: [https://web.archive.org/web/20111120222535/http://www.ugc.ac.in/policy/revised\\_finalugcregulationfinal10.pdf](https://web.archive.org/web/20111120222535/http://www.ugc.ac.in/policy/revised_finalugcregulationfinal10.pdf)] (short link: <http://bit.ly/39INmxu>) for following broken link] [http://www.ugc.ac.in/policy/revised\\_finalugcregulationfinal10.pdf](http://www.ugc.ac.in/policy/revised_finalugcregulationfinal10.pdf).

I think their problem is how do they compare one academician with another? They probably find it very difficult to come up with a country-wide, standardized & objective measure of how good a teacher one academician is compared to another. So, they probably ignore the teaching capability of the academician and lay emphasis on what they can easily measure which is the research publication output - both quantity and quality - of the academician. [For quality, I am given to understand that UGC uses impact factor [http://en.wikipedia.org/wiki/Impact\\_factor](http://en.wikipedia.org/wiki/Impact_factor)].

And this career growth measure naturally is a MASSIVE influence on the academic community. I mean, they are also human and would want more money and status. So the system, IMHO, encourages the academician to concentrate less on his/her teaching and focus instead on his/her research publication output.

Now I am not against academic research. Research is vital for society and academic research makes vital contributions to material progress of society. But my view is that there should be a proper balance between teaching and research in a University teacher. A further, perhaps, very radical view is that some academic teachers should be given the option to be only teachers and not researchers.

You wrote, "To summarize, academia focuses on things that are (a) formal and (b) original while forgetting what really matters, which is whether it's useful."

I can fully empathise with you when you wrote, prior to the summary, about how you were interested in doing something that people find useful (Orkut for e.g.). My thinking is the same and that is why I have chosen not to get involved in research focused on only producing research publications.

But then I respect the choice of academicians to do research. Their life and their choice. And, IMHO, as I said earlier, academic research does have a vital role in society.

I don't think I agree with your generalization that academia focuses on things that are original but ignores whether it is useful. Academic freedom allows academicians, as I understand it, to pursue whatever they want irrespective of whether what they are doing will be useful or not (many times the utility aspects are discovered later on). So some academicians choose to be pure theoreticians and stay distant from applications of that knowledge.

But many academicians choose to work on applied areas and there are many journals with significant impact factors which focus on such 'applied research' work. In fact, I feel some journals on topics like 'Technology for Education' look like 'research & development journals' as against a pure 'research only - no development' journal. See: <http://www.blackwellpublishing.com/journal.asp?ref=0007-1013> & <http://www.educational-software-directory.net/publications/journals>

Perhaps ELITE academia like an IIS (Indian Institute of Science) or some sections in a CS dept. in an IIT choose to focus on 'higher intellectual complexity' 'pure research' areas. Sometimes I feel it is almost like a religious monk's quest for 'purity' - pure intellectual quest of a particular knowledge area untainted by application of that knowledge area to society. But there are lots of other academic institutions in CS & IT which focus on 'applied research' areas which are aimed to doing something useful to society.

I wholeheartedly agree with your comment about iPad vs. Unix workstation. "Did your Unix workstation have more than a hundred million users who are repelled by the fact that traditional computers are too complex and don't work right in ways that people care about?" Having come from a generation where I have been dazzled when I was a young techie, by the power and attractive UI of the Unix workstation as compared to the dumb terminals of mini-computers & mainframes, I was completely bowled over by my iPod Touch a year or so ago.

My God! What a revolution in making computing devices so intuitive and so useful and so powerful! Wow!! Hats Off to Apple. Haven't tried out an iPad yet but I can imagine what a winner it will be for your Mom and millions or should I say potentially billions of people as compared to even the most powerful Unix workstation that money can buy today.

About the MCA student who should have learnt Ruby on Rails or CherryPy or ... by himself - well, what I have realized as a teacher in academia, who, btw, focused only on teaching and ignored research, is that only a few smart and motivated students have the ability to do self-learning. Most need to be taught by a teacher. That is why parents spend money and send their students to university. I mean, if the student is smart enough to learn by himself why should he come to college to learn? [Of course, he may come to/ be sent to college for the experience of college life which has its own importance.]

You perhaps do not have a teacher's view. Typically, it is once you take on that huge burden of teaching some knowledge to others that you experience the wide difference in intellect and grasping ability among

students. My experience is that it is quite easy to teach standard stuff to a smart student - the real challenge is to motivate and impart knowledge to the average student.

## **CS & IT Academia: How To Relieve Suffering of Students, Parents and Employers**

Associated blog post date: 8th Nov. 2011, link: <http://eklavyasai.blogspot.com/2011/11/cs-it-academia-suffering-students.html> , short link: <http://bit.ly/suffer-s>

I think it is universally known that Indian CS & IT academia has very poor software development skill set - as its focus is on theory and research publication skill set.

Who is suffering due to this lack of software development skill in CS & IT academia? Is it the CS & IT academics? No, not at all - as their promotions are based on research publications and not on software contributions.

The people who suffer are students, parents and employers. The software industry suffers and so the economy of India where software is a very important component, suffers. In fact, software is so pervasive in all fields today that all the industries needing software skill set suffer and not the software industry alone. In general, I think I can say that almost everybody is suffering due to this lack of software development skill set in CS & IT academia.

But are the CS & IT academics to blame? No, not at all. They are human beings like all of us. The academic system gives far more importance to research publications instead of software contributions and so they naturally focus on research publications. It typically would be a wrong decision career wise for a CS & IT academic to focus on software contributions - that is the plain and unvarnished TRUTH. Once the CS & IT academic system changes, the problem will get solved. As simple as that, IMHO.

Prof. S.S. Mantha, the AICTE chairman is a mechanical engineer by training but is also a software specialist! See: *[link is broken now in Feb. 2020]* <http://www.aicte-india.org/adcv.htm>. He has implemented e-Governance projects, provided IT expertise for IT initiatives of several departments of the Govt. of Maharashtra, and is currently the IT Expert for the Department of IT, Government of Maharashtra. He also has co-authored a book on C++ Programming (Prof. Stroustrup created the C++ programming language). So Prof. S.S. Mantha, a professor of Mechanical Engineering, seems to have a software contribution record which most CS & IT professors in the country will not be able to match!!! That is because the CS & IT professors focus on research publications instead of software contributions due to lack of significant rewards/recognition in academia for software development work.

Further Prof. Mantha does not seem to have any academic qualifications for CS & IT. He seems to be self-taught in software. And he is not alone in that regard. Here are some well known self-taught software preeminent people:

a) The Late Steve Jobs, perhaps the most famous computer technologist today, known for the iPod, iPhone and the iPad amongst other things. [http://en.wikipedia.org/wiki/Steve\\_Jobs](http://en.wikipedia.org/wiki/Steve_Jobs) [Steve Jobs was far more into hardware but did have software knowledge too.]

b) Bill Gates, co-founder of Microsoft and famous philanthropist. Here is his wiki: [http://en.wikipedia.org/wiki/Bill\\_Gates](http://en.wikipedia.org/wiki/Bill_Gates)

c) Mark Zuckerberg, famous founder of Facebook. He is only 27 years old now. Here is his wiki: [http://en.wikipedia.org/wiki/Mark\\_Zuckerberg](http://en.wikipedia.org/wiki/Mark_Zuckerberg)

All the three persons above dropped out of college before finishing their graduation. But they developed / were instrumental in developing very significant software which had a MASSIVE IMPACT on the world. I am quite sure their research publication record, if any, will not be great. So we can conclude that the software contribution/development skill set is a very different skill set from a CS & IT research publication skill set.

I think it is clear that for the dire situation to improve, CS & IT teachers must become experts in software development and have significant software contributions. Otherwise how will they be able to capably teach software development skills to CS & IT students? And for that to happen, IMHO, AICTE/UGC regulations should have two simple changes:

a) Provide promotion incentive for CS & IT teachers based on software contribution record as well as research publication record. The software contribution record should be given as much importance as research publication record.

b) Make it mandatory for CS & IT departments to show software contribution record as well as research publication record of all CS & IT teachers on its website so that prospective students and their parents, and prospective employers can make well informed choices about which college/university to join/recruit from.

Currently AICTE/UGC may be following a "one size fits all" policy with same recruitment and promotion norms for all branches of science, engineering, commerce & humanities/arts. CS & IT are very different areas with explosive growth in knowledge and in applications of that knowledge across almost every aspect of life like life-saving surgical equipment, technology enhanced education, business process workflows, e-Governance, home entertainment etc. Software Development has become an inter-disciplinary skill set with almost all branches of science, engineering, commerce as well as humanities having a need for it at some level. E.g. Language learning software is quite popular to teach/learn English.

Given this background AICTE/UGC must treat the practice oriented CS & IT branches differently from other branches like Mathematics or Physics.

I think it is high time AICTE/UGC change CS & IT academic regulations to provide promotion incentive for software contribution (development). Otherwise CS & IT students, parents of those students and huge number of employers needing software skilled professionals will continue to suffer. Note that the employers are not limited to software industry as software has permeated almost all walks of life today and so software skill set is needed in all kinds of industries and offices.

So I think I can say that the entire country will continue to suffer unless AICTE/UGC act.

## **CS & IT Academia: Inform Students About Software Contribution Record of Faculty**

Associated blog post date: 4th Nov. 2011, last updated on 8th Nov. 2011, link:

<http://eklavyasai.blogspot.com/2011/11/cs-it-academia-inform-students-about.html> , short link: <http://bit.ly/inform-s>

Some of the proposals I have outlined in previous blog posts may have some implementation challenges, even if the proposals are found interesting by CS & IT academic administrators.



Here is a simple and, IMHO, easily implementable proposal to improve software development skills in CS & IT academia.

AICTE & UGC should make it mandatory for CS & IT departments to put up the software contribution (development) record of each faculty on its website. Now, typically, only the research interests and, possibly, research publication record of the faculty are put up.

The software contribution record could have following information for each software contribution:

1. Short description of the software including development platform (Operating System, Software tools used etc.)
2. If the software is free or open source then a download link from where the software & documentation can be downloaded and used/verified by anybody.
3. If the software is proprietary then a link to the website of the company for which the software was developed where the software details are mentioned. Ideally that page should have a credits menu/link which should list the faculty as a contributor.
4. Role faculty played - Architect, Requirements Analyst, Designer, Developer, Tester etc.
5. Duration of involvement of faculty with the software development.

Besides such individual faculty software contribution record, a consolidated department software contribution record should also be put on the website.

Such information would allow students, parents and employers to get some idea of the software development skills of the faculty & department besides the research skills.

If a CS / IT department has only research publication record with very limited or no software contribution record then students, parents and employers who are interested in software development skill set can make an informed choice to not study in/recruit from that department. Instead students, parents and employers who are interested in research can make an informed choice to study in/recruit from that department.

Okay, but how will this proposal improve software development skills in CS & IT academia? Well, the reality is that the overwhelming majority of students, parents & employers are interested in software development skills. So CS & IT departments which have zero or limited software contribution record will, over time, lose out to CS & IT departments which have a good software contribution record. That will motivate CS & IT academia to improve its software development skills.

---

Given below is a response I made to one comment in the blog post (please visit blog post to view that comment):

Eklavya Sai Maalik (Ravi S. Iyer) wrote on November 7, 2011 at 2:47 PM

Thanks for your views.

You wrote, "The companies that hire know they can turn a Chemical/Civil Engineer into a IT professional". IMHO, companies take on the burden of education of freshers because of CS & IT academia's failure to teach them practical skills (design & code) properly. If companies had a choice of freshers who are industry-ready software professionals they will surely opt for them as then they will save the costs of company-paid-education.

You wrote, "The students in turn know they will get the job and can keep it with little effort and they need not have to be really good programmers coming out from college." Do all CS & IT students get a job so easily? Can software professionals in all companies keep their jobs with little effort? Is it so simple? I feel there are plenty of software industry jobs which need very good design & code skills. Further it takes significant effort to create good software.

You wrote, "Just like it is mandatory for doctors to practice in rural areas, IT professionals must spend time with academia. The government should enforce such commitments from the software industry." Force does not work in a democracy. And, thank God for democracy. Indian CS & IT Academia which must be receiving significant amount of tax payer money, must make efforts to welcome software industry professionals to teach. Unfortunately AICTE/UGC regulations for recruitment and promotion of teaching staff are such that industry expertise is not given any real importance unless the person has a PhD. So it is very hard for non-PhD industry professionals to consider changing their career from industry to academia. Without a PhD they can join only as an Asst. Professor which is the entry level. A person with significant industry experience would feel odd to be clubbed with entry level teachers. He/she would expect to be appointed as an Assoc. Professor. That needs a PhD. Once again, the academic "PhD club" mindset comes into play.

## **CS & IT Academia: Profs. May Not Like Idea of Software Contribution Record**

Associated blog post date: 8th Nov. 2011, link: <http://eklavayasai.blogspot.com/2011/11/cs-it-academia-profs-may-not-like-idea.html> , short link: <http://bit.ly/profs-not>

This post captures a slightly edited email exchange with a friend on the "CS & IT Academia: Inform Students About Software Contribution Record of Faculty" post (given earlier in this book).

Friend wrote: It is a clever idea and tries to work around the problem but it seems like a somewhat superficial solution to a deeply ingrained problem.

Eklavya Sai (Ravi S. Iyer) responded: I entirely agree that it is not a proper/complete solution to a deeply ingrained problem.

Friend wrote: I don't think profs will go with the suggestion since they think programming expertise isn't important in the first place and therefore don't want to be judged by this irrelevant (to them) criterion.

Eklavya Sai (Ravi S. Iyer) responded: Vital point. I read somewhere, in a recent newspaper issue I think, about HRD Minister Shri Kapil Sibal talking about independence of academia from government (interference) during some IIMs related discussions.

So profs will reject it as it does not suit them. Some may realize the importance of programming/software development but will not say so in public when such a measure is suggested.

But I am given to understand that students & parents at counseling (engg. college admission) time do question college/university representatives about the placement record. If placement record is poor then they tend to shy away from that college/university. So college/university managements give value to placement and as they pay the salaries of the academics, the academics have no choice but to listen to them.

If AICTE/UGC feel that this "software contribution record" is worth trying out and agree (perhaps under people pressure), then profs. may have no choice but to go with it.

Friend wrote: Students and parents have no idea what makes an engineer or a prof a good one any more than they know what makes a painter good.

Eklavya Sai (Ravi S. Iyer) responded: I am given to understand that many students & parents do put a lot of questions at counseling time. I think the high college education costs make it somewhat like an investment and so I think they do try, with their limited knowledge, to assess where best to invest their time & money. If "software contribution record" is available then they may use that like they use placement record, as a parameter to decide their choice.

Friend wrote: Besides ideas that require the network effect to be useful are hard to get off the ground, unlike ideas that are useful to their practitioners even if others don't go with them.

So, sorry, but I'm not bullish on this one.

Eklavya Sai (Ravi S. Iyer) responded: I can understand. To be honest, I know that we are up against a solid wall.

Friend wrote: We can still give it a shot, though.

Eklavya Sai (Ravi S. Iyer) responded: That's exactly what I feel. I mean, it is like negotiating a tough deal. If MHRD and NASSCOM agree that something needs to be done, then AICTE/UGC admin. profs. may have to accept something. This may be the easiest thing to push through.

Even if it does get pushed through, whether it will really be adopted in the proper spirit ... and so succeed is another thing altogether. For that we have to experiment and wait & see.

I guess all we can do is raise the problem, suggest some easily implementable solutions and use 'amicable & peaceful' means to make them (MHRD/AICTE/UGC) at least acknowledge these issues & suggested solutions. After that it is completely out of our hands.

## **Publicly Funded Higher Education Institutions should put up Detailed Course contents on the Internet**

Associated blog post date: 7th Apr. 2014, last updated on 8th Apr. 2014, link:

<http://eklavyasai.blogspot.com/2014/04/publicly-funded-higher-education.html> , short link: <http://bit.ly/pub-funded>

Given below is an email that I plan to send shortly to top persons associated with Indian academic/higher education funding, oversight and administration. [Update: Mail sent on 7th April 2014.]

Dear sir/madam,

In the computer science/information technology (CS/IT) academic field some US universities have done an outstanding public service by making available freely on the Internet the following for their courses:

- Course structure
- Course book(s) - if the course is based primarily on it/them
- Reference books and other resources, if any
- Teaching material (could be prepared by the faculty or be a re-use of external publicly available material with clear attribution or as an external link)
- Assignments

Here is an example of such detailed course content: Advanced Programming in the UNIX Environment – [link is broken in Feb. 2020] <http://www.cs.stevens.edu/~jschauma/810/>

As a teacher of over ten different computer programming (lab.) courses (as honorary staff/honorary faculty/visiting faculty) in a deemed university in Andhra Pradesh, India from 2003 to 2011, I found such course content to be very useful and was inspired by such examples and tried to, whenever I could, do the same for my courses on the university Intranet. [Now I am in the process of putting up the course material of most of these courses on the Internet here: <http://raviiyerteaches.wordpress.com/>, just in case it may be of use to others on the Internet.]

It seems to me that most CS/IT departments of Indian higher educational institutions generally do not follow such a practice. Some have course content pages (e.g. some of the IITs) but the information contained is minimal and does not come close to what I have mentioned above about some US university course pages. Yes, we have NPTEL, <http://nptel.ac.in/>, but that is a single national level portal which may find it difficult to accommodate various types of teaching methodologies and levels of rigour across the huge variety of educational institutions in the country.

Given this situation, I have a suggestion that publicly funded higher educational institutions in India should strongly encourage the faculty of these institutions in CS/IT departments and other departments as well to put up their course material in detail on their institution website so that it becomes available to any interested person in India (and abroad) over the Internet. I think it will be a wonderful return to the public on public money investment in publicly funded higher educational institutions of India. University assessment organizations like NAAC and NBA as well as prospective students (and their parents) can then look up these course pages to get some idea of the methodology and rigour used by the faculty to teach these courses.

As of now, it is extremely difficult for outsiders including prospective students (and parents) as well as, I presume, university assessment organizations like NAAC and NBA to get a decent feel of teaching standards in most Indian higher education institutions. This sort of transparency and accountability for teaching duties of faculty especially in publicly funded higher educational institutions of India may do wonders for improving the teaching standards of higher education in India.

Regards

Ravi S. Iyer

Software Consultant

---rest of email signature snipped---

---

[Mail update (below) sent on 8th April 2014]

A couple of clarifications:

1. By publicly funded higher education institutions I mean government (tax payer) funded higher education institutions.
2. The last sentence of the main body of the previous mail would be better expressed as follows: The above mentioned suggestion (faculty putting up detailed course material on the Internet) may result in the sort of transparency and accountability for teaching duties of faculty especially in publicly funded higher educational institutions of India that may do wonders for improving the teaching standards of higher education in India.

# My suggestions related to skill development in Indian higher education on mygov.in discussions on New Education Policy

Associated blog post date: 20th Feb. 2015, link: <https://eklavyasai.blogspot.com/2015/02/my-suggestions-related-to-skill.html> , short link: <http://bit.ly/my-sugg>

First, I would like to state my privacy and ethics policy on sharing publicly, matter related to mygov.in discussions.

- a) I am free to put up on my blog for public viewing, the comments I make on mygov.in.
- b) I should not and will not share any content of mygov.in portal that is private to logged in users and that is not contributed by me [unless I seek and get specific permission, which I don't think I will have any occasion to do so.]
- c) I am free to put up links to content of mygov.in website which are accessible to the public

--- end privacy and ethics policy on sharing publicly mygov.in discussions ---

What is the New Education Policy that the Indian central (federal) government is working on, and is inviting participation from concerned Indian citizens like students, parents and teachers? This link gives a good idea: <http://mygov.in/new-education-policy-group.html>. A short extract from it:

The National Policy on Education was framed in 1986 and modified in 1992. Since then several changes have taken place that calls for a revision of the Policy. The Government of India would like to bring out a National Education Policy to meet the changing dynamics of the population's requirement with regards to quality education, innovation and research, aiming to make India a knowledge superpower by equipping its students with the necessary skills and knowledge and to eliminate the shortage of manpower in science, technology, academics and industry.

--- end extract ---

Related to skill development in higher education I made the following suggestions (split into multiple posts due to post size limitations; I chose not to upload a pdf file) on the mygov.in portal (slightly edited):

In technical higher education like Computer Science and Information Technology, in my view, these are the problems and suggested solutions:

a) Lab. courses which are meant to provide skills are not given much importance. Students are not examined strictly (it is rare for students to be failed in lab. courses), and so they tend to take it easy with lab. courses. Further, faculty are not given any career growth incentive for teaching lab. courses well. So lab. courses are usually dumped on junior faculty.

Solution: Do strict evaluation of lab. courses, failing students where necessary. Ensure career growth incentive for lab. course teaching faculty.

b) Industry experience is not given much value by UGC/AICTE recruitment and promotion norms. So industry experienced persons do not have much incentive to move from industry to academia as teaching faculty. Non-industry-experienced academics in fields like Computer Science and Information Technology usually lack thorough knowledge of practical side of the field.

Solution: Improve practical skills level of teaching faculty in technical higher education by making it easy for experienced and accomplished industry professionals to move to academia on regular academic pay scale and designation.

c) In general, I am given to understand that in medicine the teaching faculty are themselves practitioners of medicine in "teaching hospitals". However, in technical fields like Computer Science and Information Technology, typically the teacher is not a practitioner of the field! So then how can the practical skills be taught properly to students?

Solution: Strongly encourage via career growth incentives, teaching faculty in technical higher education to be practitioners of the associated field. Measures of their competence as a practitioner can be evolved over time, based on artifacts/prototypes/products developed by the teaching faculty which should be peer-reviewed to ensure good quality output and rejection of poor quality/fraudulent output. Specifically, in the Computer Science and Information Technology field, a software contribution record can be used as a measure of competence of the teaching faculty in the practice of software development.

Note that similar/same suggestions were made to a UGC committee as mentioned in this post of mine: Suggestion to Committee on attracting and retaining talented and quality manpower in Teaching Profession, <https://eklavyasai.blogspot.com/2015/09/suggestion-to-committee-on-attracting.html> (short link: <http://bit.ly/suggestion-t> ), 11th Sept. 2015. Relevant extract from the post is given below:

Based on this notice in the UGC website, [http://www.ugc.ac.in/pdfnews/5390499\\_quality-teaching-profession.pdf](http://www.ugc.ac.in/pdfnews/5390499_quality-teaching-profession.pdf), I sent today (11th Sept. 2015) the following email to the person mentioned in the notice, "Dr. Sunita Siwach, Deputy Secretary, UGC" <ssiwach.ugc@nic.in>:

Proforma to invite suggestions/views for consideration of Committee constituted by the Central Government to examine issues related to attracting and retaining talented and quality manpower in the Teaching profession

Name and Address of the Stakeholder: Ravi S. Iyer, ---snip---

E-mail address: ---snip---

Name of the vice-chancellor of institution: NIL; I am writing as a software consultant who is currently not associated with any university or industry company.

Tel. No.: ---snip---

My Suggestion:

Related to skill development in technical higher education in fields like Computer Science and Information Technology, in my view, these are the problems and suggested solutions:

*[What follows in the blog post: Suggestion to Committee ... is same/very similar to problems & solutions given earlier in this article and so I am not repeating it.]*

--- end extract from blog post: Suggestion to Committee ... ---

# **How MHRD (& DIRECTOR IIT) disposed of my "Serious Systemic Problems in Indian CS IT Academia" grievance dated Nov. 2011, DOING NOTHING REALLY**

Associated blog post date: 15th Sept. 2015, link: <https://eklavyasai.blogspot.com/2015/09/how-mhrd-director-iit-disposed-of-my.html> , short link: <http://bit.ly/how-mhrd>

I had put up my complaint as a grievance in Nov. 2011 at the following portal, <http://pgportal.gov.in/>, whose description is as follows:

PORTAL FOR PUBLIC GRIEVANCES, Brought to you by Department of Administrative Reforms & Public Grievances, Government of India

I received an email (and SMS) today (15th Sept. 2015) that my grievance has been DISPOSED. The status of the grievance could be checked at the portal by (mainly) giving the registration number. [No login is required; so anybody can do it.] I have given below the Grievance Status. [BTW MHRD stands for (Union/Federal) Ministry of Human Resource Development, <http://mhrd.gov.in/>, which is the main ministry handling education policy and public (tax payer) funds disbursement to UGC/AICTE and other such institutions which disburse it to suitable educational institutions across the country.]

Registration Number : DSEHE/E/2011/01988

Name Of Complainant : Ravi S. Iyer

Date of Receipt : 10 Nov 2011

Received by : Department of Higher Education

Forwarded to : DIRECTOR IIT

Contact Address : ROOM NO 516 C

SHASTRI BHWAN

NEW DELHI 110001

Contact Number : 23073241

Grievance Description : Dear sir/madam, I had sent a mail on Oct. 28th 2011 to Hon'ble ministers, MHRD and others on "Serious Systemic Problems in Indian CS IT Academia" (see attached document) but have not received any response so far. I have written another post on the issue with some concrete suggestions for a solution, entitled "CS IT Academia: How To Relieve Suffering of Students, Parents and Employers" here: <http://eklavyasai.blogspot.com/2011/11/cs-it-academia-suffering-students.html> I would be very happy to receive some acknowledgement of receipt of this grievance suggested solution, and your esteemed ministry's views on the matter. I also offer my services, in case your esteemed ministry is interested, in a part-time, free Seva capacity to help arrive at some solution to this serious problem for the entire country. Regards Ravi Iyer Software Consultant Puttaparthi, Andhra Pradesh e-mail: [ravi@raviiyer.org](mailto:ravi@raviiyer.org) web: [raviiyer.org](http://raviiyer.org) - Individual Capacity, Not-For-Profit Effort Service to Society is Service to God

Current Status : CASE CLOSED

Date of Action : 15 Sep 2015

Details : The matter does not pertain to MHRD. Even otherwise it is not a grievance.

--The following were provided by me (Ravi S. Iyer) today (15th Sept. 2015) on the status page --

Feedback Rating: Poor [Lowest rating]

Feedback comments: [The website accepted the comments below but does not show it on the status window. It does show the above feedback rating though. Also note that a lakh is an Indian term for one hundred thousand. So 'lakhs of students' means 'hundreds of thousands of students'.]

As explained in the doc. attached to my grievance (and available on the net here:

<http://eklavyasai.blogspot.in/2011/09/cs-it-academia-serious-systemic.html>), "The unfortunate reality of the vast number of CS & IT departments of universities & colleges in the country is that the Professors of these departments are strong in theory and research publications but not so strong in practical areas like Software Design and Programming or Coding. Naturally the students graduating out of CS & IT academia also end up having somewhat strong theoretical and research-oriented skills but being weak in Software Design and Programming."

The doc. also explains that AICTE/UGC policies do NOT encourage CS/IT academics to know programming well, and that these policies need to be changed. Further, as explained in the same document, MHRD is the govt. agency that gives TAX PAYER money to AICTE/UGC, and so MHRD is answerable to the Indian public for AICTE/UGC policy failures affecting lakhs of students & parents in the country.

So I completely disagree with MHRD's reasons for closing this grievance. Not only is my grievance a valid grievance, it also pertains to MHRD.

The grievance status states that it was forwarded to DIRECTOR IIT. What response did MHRD get from him/her? Why has that response not been shared on this status? Is the DIRECTOR IIT not answerable or accountable for AICTE/UGC policies that he/she may have been instrumental in formulating, and that affects the lives of lakhs of students & parents in the country?

--- end Feedback comments ---

=====

I sent the above blog post contents as an email to the Hon'ble minister for Human Resources Development, Smt. Smriti Irani (on 15th Sept. 2015).

## **About me not being Ph.D. or qualified in Computer Science but yet teaching lab. courses and being tech. consultant in an Indian deemed university in the past**

This post is based on part of my blog post: <http://eklavyasai.blogspot.com/2016/08/about-me-not-being-phd-and-yet-teaching.html> (short link: <http://bit.ly/about-me-not> ), first posted on 20th Aug. 2016 and last updated on 24th Feb. 2020.

I thought some readers who are not knowledgeable about Indian Computer Science academia may wonder why I was teaching lab. courses in the Indian deemed university (DU) though I am not a Ph.D. in Computer



Science or otherwise educationally qualified in Computer Science. So I thought of putting up this post explaining the situation.

Well, I did not join the DU for an academic career. I retired from commercial work in Aug. 2002 and moved to a spiritual town in Oct. 2002 with an objective of focusing on my "spiritual career" there if the spiritual master there was willing to accept me for such a role. To cut a long story short, in January 2003 my offer to provide Seva (free service offered with a spiritual mindset) to Dept. of Mathematics & Computer Science in the DU was accepted, based on my industry experience of 18 years in international software industry (and not based on my educational qualification). Note that Bill Gates, co-founder of Microsoft, Late Steve Jobs, co-founder of Apple, and Mark Zuckerberg, co-founder of Facebook, all three of these iconic software industry figures did not complete their graduation (forget about PhD). So I am more qualified than them educationally as I am a graduate!!! What matters is their industry achievements/experience. Academia is not the only place one can learn software development. The software industry is a superb place to learn and practise software development, perhaps far better than Indian Computer Science academia as it is now.

Educational qualification wise, I am B.Sc. (Physics) from Bombay University (passed out in 1983). I had joined M.Sc. (Physics) in Bombay University but due to money problems decided to drop out of my M.Sc. after around six months. A few months after dropping out of M.Sc., in March 1984 I was into the software industry as a trainee programmer in my first software company, Datamatics, Mumbai. My biodata is given in an Appendix of this book and can also be viewed here:

<https://ravisiyer.files.wordpress.com/2020/02/202002-ravisiyerbiodata.pdf> (short link: <http://bit.ly/rsi-biodata>), and details about my software industry experience is also given in an Appendix of this book, and can also be viewed here: <https://ravisiyer.files.wordpress.com/2011/09/raviiyerindustryworkexperience.pdf> (short link: <http://bit.ly/rsi-workexp>).

At the time I joined the DU I had no demands whatsoever about designation. But I think the official designations provided by Principal, of the DU campus, to me of Honorary Staff, Honorary Faculty and Visiting Faculty over my nine year stint, were fair. Later, as I studied UGC norms for such matters, I realized that the right designation for me from UGC norms point of view would be "Visiting Faculty". [In fact, in that DU campus, they used to use a term "Regular Visiting Faculty" to describe people like me who were staff doing teaching function (for FREE typically) for the whole academic year, to differentiate us from other Visiting Faculty who would visit the university for a few weeks in a year, typically (for FREE).]

Now, I could have chosen to do Computer Science research and publish academic papers on it, which would have made me PhD equivalent in Computer Science (5 published papers of certain degree of standing is considered equivalent to Ph.D. as per UGC norms, if I recall correctly). As part of my Seva, I used to act as technical consultant for M.Tech. projects and have published two conference papers as a co-author along with the student and another Visiting Faculty from the USA who was a regular academic Professor. I did that work only to help the student and the department (as it counts towards the department research profile).

If I wanted to publish these 5 papers to meet UGC requirement mentioned above for PhD equivalence, I surely could have done it. Please excuse me about blowing my own bugle, but I have to say that it would not have been a problem for me at all. But I had NOT moved to the spiritual town for an academic career in Computer Science! I had come for a "spiritual career". So I consciously chose not to spend time on doing Computer Science academic research and publishing papers in it (except in the above mentioned two cases to help the students & dept.) My expertise was on software development side - implementation side. I preferred to limit myself to that role which was perfectly OK for my role of teaching software development lab. courses and being technical consultant on software development aspects of M.Tech. (Computer Science) projects. [I also played a part-role of Software Lab. Manager for the first 5 years of my free service (Seva) stint in the DU.]

My main interest in using my spare time after my teaching and technical consultant free service in the DU, was in spirituality & religion. I preferred to do a lot of reading in that area instead of doing Computer

Science research. I think I have read far more books from the DU campus library on spirituality & religion, than I have read books from that library on Computer Science or any other subject/field!!!

**Views of Top Leaders (Country  
Presidents/Prime Minister/Ministers/US  
Senators) on Education**

# Education and IT related parts of President Mukherjee's address to Parliament; My comments

Associated blog post date: 9th Jun. 2014, link: <http://eklavyasai.blogspot.com/2014/06/education-and-it-related-parts-of.html> , short link: <http://bit.ly/prez-speech>

Here are some extracts and comments of President Mukherjee's speech to Parliament today (here's the full speech - a 14 page pdf document, [link is broken in Feb. 2020]  
[http://www.pmindia.gov.in/President\\_Address.pdf](http://www.pmindia.gov.in/President_Address.pdf)):

12. India is the world's oldest civilization. Today it is also a country with the largest population of the youth. We must equip and nurture our youth with the right kind of education, skill-set and opportunity to reap this demographic dividend. My government will strive to transition from Youth Development to Youth-led Development. It will set up Massive Open Online Courses and virtual classrooms. It will formulate a National Education Policy aimed at meeting the challenges posed by lack of quality, research and innovation in our educational institutions. We will set up IITs and IIMs in every state. In order to empower school teachers and students, a national e-library will be established. With the motto of "Har Haath Ko Hunar", my government will strive to break the barriers between formal education and skill development, and put in place a mechanism to give academic equivalence to vocational qualifications. With the goal of Skilled India, my government will also launch a National Multi-skill Mission.

*[Ravi: I am so glad to see that education has been given significance in this gov't's plan. Really great to see MOOCs being mentioned in the speech. MOOCs are the growing waves in higher education worldwide and seem to be going from strength to strength. India MUST get on to the bandwagon perhaps with a tie-up with leading players like edX and CourseEra.*

*"National education policy" - very interesting. But they must open it up to the public during the discussions stage itself rather than leave it to a small coterie of top academics and industry experts to make the policy, which is what seems to have been the case in the past. Use the Internet to involve all stakeholders including the vital students and parents stakeholders as well as junior faculty who have to deal with the teaching loads.*

*"academic equivalence to vocational qualifications" - wonder what that exactly means ... National Multi-skill Mission. Interesting. But one needs more information about them to comment meaningfully.*

*IITs and IIMs in every state - I am somewhat disappointed by no mention of UGC & AICTE specifically (but National Education Policy may involve UGC & AICTE) which cater to the vast majority of higher education students in the country. There seems to be a belief that bringing an IIT and/or IIM to a state will magically improve its higher education. It may do that for the brightest/elite among the youth of that state. But how about the other students who can't get into the IIT/IIM? They are the vast majority. And it is UGC & AICTE regulated institutions that cater to them. The govt. must explore ways to improve such institutions on a priority basis as it affects a vast number of youth, and not give so much importance to IITs and IIMs that UGC & AICTE institutions get treated as unimportant.]*

...

The government will especially strengthen measures to spread modern and technical education among minority communities and a National Madarsa Modernization Programme will be initiated.

*[Ravi: That's excellent.]*

...

22. E-governance brings empowerment, equity and efficiency. It has the power to transform peoples' lives. The backbone of my government's new ways of working will be a Digital India. IT will be used to drive re-engineering of government processes to improve service delivery and programme implementation. We will strive to provide Wi-Fi zones in critical public areas in the next five years. My government will rollout broadband highway to reach every village and make all schools e-enabled in a phased manner. Technology will be used to prepare our children for a knowledge society. The National e-governance plan will be expanded to cover every government office from the centre to the Panchayat; to provide a wide variety of services to citizens. Emerging technologies like Social Media will be used as a tool for participative governance, directly engaging the people in policy making and administration.

*[Ravi: Fascinating! Shri Modi is strongly pro-technology and pro-e-governance. It will be very interesting for an IT guy like me to see how these plans work out over the coming months and years.]*

...

36. Our rich cultural heritage is the very foundation on which rests the unity of our diverse nation. Indian languages are repositories of our rich literature, history, culture, art and other achievements. My government will launch a national mission "e-Bhasha" that will develop digital vernacular content and disseminate our classic literature in different languages. My government will also provide the required resources for the maintenance and restoration of national heritage sites.

*[Ravi: That's great. Hopefully it will do a decent job, if not a good job.]*

...

38. My government recognises the central role of Science and Technology in raising the quality of life. It will encourage and incentivize private sector investments, both domestic and foreign, in science and technology and in high-end research aimed at nurturing innovation. My government will build world class research centres in the fields of nanotechnology, material sciences, thorium technology, brain research, stem cells, etc. The government will also establish institutes of Technology for Rural Development and a Central University of Himalayan Studies.

--- end President speech education and IT related extracts and comments ---

*[I thank former President Mukherjee of India and have presumed that the President's office will not have any objections to me sharing the above small extract(s) from their website in this book, whose ebook is freely downloadable/viewable by all, and whose paperback book is not-for-profit for me (author). This book does not have any financial profit motive whatsoever for the author.]*

## **Education and IT related excerpts from PM Narendra Modi speech in Lok Sabha on June 11th 2014**

Associated blog post date: 13th Jun. 2014, link: <http://eklavyasai.blogspot.com/2014/06/it-and-education-related-excerpts-from.html> , short link: <http://bit.ly/pm-speech>

This speech was, if I am not mistaken, the concluding speech on the motion of Thanks for the President's speech on June 9th (earlier article in this book is on it). Hon'ble Prime Minister Shri Narendra Modi mentioned in his speech that around fifty speakers (from various political parties) had spoken on this motion.

I could not locate a full text (translated to English) of this speech of Shri Modi so far. This link gives a lot of points from the speech: <http://www.ndtv.com/article/india/narendra-modi-speaks-on-president-s-motion-of-thanks-in-parliament-highlights-539796>.

Some excerpts of the speech and associated timestamps, mainly related to education and information technology, transcribed from this speech video, <https://www.youtube.com/watch?v=9vUMYUaaY40>, 1 hr 1 min 30 secs, are given below. These excerpts are a mix of full translation of Hindi sentences as well as only mention of key points.

### **Around 11:35 - Government's first responsibility is towards the poor**

If the rich need to teach their children they can send them to any school in the world/hire any teacher in the world. If somebody in the home of the rich fall ill, many doctors can come to his home. Where will the poor go? In his destiny (*naseeb*) he has (only) government schools and government hospitals. Therefore it is the first responsibility (*daayitva*) of the government that they listen to the poor and live for the poor. If we do not run the government machinery (*karobaar*) for the poor, for the benefit of the poor, then this country's people (*jantha*) will not (never) forgive us.

### **Around 13:25 - Empower the poor to fight poverty through education and liberation from blind belief**

(Hindi: *Garib ko garibi se bahar lane ke liye, Uske andar wo taakath lani hai taaki wo garibi ke saath jujh sake. Garibi ke khilaaf ladaai ladne ki sabse bada (aujaar?) hota hai - shiksha. Garibi se ladne ka sabse bada saadhan hota hai - andh shraddha se mukthi. Agar hum garibi me - jo aaj andh shraddha ke bhaav pade hai, ashiksha ki avastha padi hai, agar usme se use bahaar lane me safal hote hai to is desh ka garib kisi ke tukdo par palne ka iccha nahi rakhta hai. Wo apne bal bute par apni duniya khadi karne ke liye taiyaar hai. Sammaan aur gaurav se jeena garib ka swabhaav hai. Hum iski us moolbhoot taakath ko pakad karke agar usko bal dene ka prayaas karte hain - Aur isliye sarkaar ke yojnaaye garib ko garibi se bahaar aane ki taakath de. Garib ko garibi ke khilaaf ladaayi ladne ki taakath de. Aur shaashan ki saari vyavasthayein garib ko shashakth banane ke liye kaam aani chaahiyein. Aur saari vyavasthaon ka antim nateeja us aakhri shor par baithe huen insaan ke kaam me aaye, us disha me hoga tab jaakar hum uska kalyan kar payenge.*)

English translation of above (there may be minor inaccuracies as my Hindi knowledge is of middling kind): To bring the poor out of poverty, he has to be given that strength, that power that will enable him to fight poverty. To fight poverty, the biggest weapon is - education. The biggest means to fight against poverty is liberation from blind belief. If we, within/among the poor who today are (trapped) in an environment of blind belief, in a state of being uneducated, are successful in removing them from that state then the poor of this country (do/will) not wish to live on the crumbs of others. He is ready to create his own world (life) based on his own strength. It is the nature of the poor to live with dignity and respect. If we catch this root strength of his (of the poor) and make efforts to give (more) power to it - and so the plans/projects of the government should give strength to the poor to come out of poverty; give strength to the poor to fight against poverty. And the entire government administration should be of help/use to empower the poor. And the end result of all administration should be in the direction of being useful to the human on the final (distant) shore (frontier), then we will be able to ensure his well being (serve him/serve the poor on the final frontier).

### **Around 15:58 - Providing best teachers to (rural) villages using long distance education technology**

We have used one word - Rurban, for the progress of villages, which I saw in the speech of the president. (Hindi: *Jahaan suvidha shehar ki ho aatma gaon ki ho*) English: Where the facilities should be of the kind in cities but the spirit (character) should be of the kind in villages. ... The noble character (the noble spirit) of the villages of our country is a priceless legacy which we must preserve. But will we deprive our villagers of modern facilities? And I am saying this with experience that when villages are given modern facilities they are making more contribution to the country's progress. If villages have 24 hour electric power supply, if villages have (Internet) broadband connectivity, (if) the children of villages get excellent education ... (If) we

accept that we do not, (as of now), have good (best) teachers in our villages but today's science (and technology) gives us the full power of long distance education. (Using) the best of the best teachers sitting in cities we can teach the children sitting in villages (of the final frontier). Why should we not use the power of satellites, use modern science (and technology) for the education of poor children? If we can bring a change in the life of villages then nobody would want to leave their villages. What does the youth want? If he/she gets employment (in the village) he/she would prefer to stay with their parents. Can we not create industries in villages? ... Agro-based industries ...

### **Around 19:40 - Agricultural universities' challenges; traditional farmer has to be changed to modern farmer**

We have so many agricultural universities. A lot of research is going on. But the unfortunate thing is that what is in the lab. is not on the land. The journey from the lab. to the land, till we do not make (that journey) ... There is a need to change the farmer from traditional farmer to modern farmer. ... Gujarat's soil health card experiment ... (govt.) came to know of inappropriate crops sowed by the farmer for the soil he has ... the farmer came to know (was informed by govt.) of the flaws of his approach and was able to consider better approaches ... Agriculture university students of our country can, during their (summer) vacation (in their home villages/towns) do soil testing in the school laboratories in their villages (or towns). The students will get paid, the school will get paid, (soil testing data will be made available) - can this not become a people's movement?

### **Around 22:42 - Indians recognized for Information Technology in the world; Yet we don't have real time data on Agri products**

Today it is unfortunate in our country. Such a big technology came into being - Information Technology. (Hindi: *Duniya mein hum chaaye huve rahein. Software engineer ke roop mein hamari pehchaan ban gayi. Lekin aaj hamare desh ke paas agro product ka real time data nahi hain.*) English: We are spread out in the world (in IT). We Indians are now viewed/recognized as software engineer(s). But today our country does not have real time data on agro products. Can we not do that? ... it will help in price control.

### **Around 29:30 - Very big need for skill development**

The whole world in the coming days has a very big need for skilled labour force/manpower. Those who know this knowledge area (*shastra*) know that the whole world needs skilled manpower. ... So skill development is an important agenda for all countries including the well developed countries. Our country's primary area (*prathmikta*) should be skill development. ... (Hindi: *To hame srameva jayate is mantra ko (charitartha) karna hoga*) English: So we will have to (adopt) the mantra of work/effort alone triumphs. ... From an image of 'scam India' we have to make the image 'skill(ed) India'. ... So we have created a new ministry for entrepreneurship and skill development. Our country has a misfortune. If we ask somebody what have you studied, he says I am a graduate or MA or a double graduate. We like hearing that. ... (talks of a book he read during childhood with a real life account of a graduate looking for a job. When the graduate is asked what do you know, the person says he is a graduate but does not answer the question i.e. does not say what he knows. He is told OK you are a graduate but what do you know. The person repeats that he is a graduate - this sequence is repeated one or two times more). (Hindi, around 32:07: *Hum is baat se anubhav kar sakte hain ki jindagi ka gujara karne ke liye haath me hunar hona chahiye sirif haath me certificate hone se baath (nahi banthi)*). English: From this we can (deduce) that to make a living we must have a skill in hand, just having a certificate in hand will not do.

And so will have to give emphasis on skill development. But we also have to give a social status for the skilled worker. (He gives an example of a 7th standard student leaving the village school and enrolling into a skill development course. But people do not give value to him - they treat him as a 7th standard educated student and ask him to leave.) We will have to bring in equivalence (to academic qualifications) for these (skilled courses). In Gujarat I tried an experiment. Those who do two years ITI course, I made them equal to 10th standard (passed/qualified) persons. Those who came after 10th standard (into ITI course?) were made

equal to 12th standard. (For these ITI qualified students) doors were opened for studying (enrolling) in Diploma or Degree programmes. He was 7th pass but he can go upto degree now - the pathway was opened up. We will have to make new decisions with a lot of courage (*himmat*). If we want to give emphasis to skill development then we will have to give social standing for that.

... Today the whole world needs teachers. Can *Hindusthaan* (India) not export teachers? If we export Maths and Science teachers - if a businessman (*vyapaari*) goes abroad then at the most he will come back with Dollars - but if a teacher goes abroad he will bring back a whole generation along with him, this is the power (of teachers). [Applause.]

### **Around 38:34 - Progress (*vikaas*) should be a people's movement (*jan aandolan*)**

We should make progress (*vikaas*) a people's movement (*jan aandolan*). If we make progress a government programme then (it won't work). But if we make progress a people's movement - in this country, every person must feel - a teacher also must feel, when I teach, I teach for the nation; even if a worker is carrying luggage in a railway station, he should feel that he is doing it for the nation, if a farmer is working on his fields, he should also feel he is working for the nation. Once again, in this country, a movement (*aandolan*) of good governance (*suraajya*) is required. ... And it is not that it is impossible. ... It is possible to create this feeling. (Hindi: *Aaj bhi hamare desh me desh bhakti ki koi kami nahi hai.*) English: Even today, in our country, there is no dearth of devotion to the country. ... (Hindi: *Aavashyakta hain ki ham nethruthva pradaan kare. Yahan baithe huen sabhi samarthyaavaan hain. Jan-jan ko jagane ki wo taakath rakhte hain - (us) taakath ko lagane ki jaroorath hain.*) The need is that we (parliamentarians) should provide leadership. All the people seated here (parliament - lower house) are capable people. We have the power to rouse the populace - that power is needed to be used.

### **Around 45:04 - Duryodhan quote - I know Dharma but it is not in my DNA!**

[Ravi: *The following two excerpts are not particularly related to education or IT. However, I felt it was pretty noble stuff and so educative in that sense. So I have included it below.*]

(Hindi: *Ek baar Duryodhan ko poocha gayaa ki bhai ye dharm aur adharm, sathya aur jhooth, tumko samajh hai ki nahi hain. To Duryodhan ne ye baath .. Usne kaha Jaanaami dharmam nachame pravratni - main dharm ko jaanta hun lekin ye meri pravratni nahi hain. Sathya kya hai mujhe maaloom hain, achcha kya hain mujhe maaloom hain lekin woh mere DNA mein nahi hain.* [Laughter.]) English: Once Duryodhan (the arch-villain in the great Hindu epic Mahabharatha) was asked whether he understood dharma (righteous action/living) and adharm (unrighteous action/living), truth and falsehood. Duryodhan said that he knew dharma but (following) dharma was not in his nature. I know what is truth and what is good, but that is not in my DNA.

### **Around 46:29 - Seniors' blessings will save govt. from arrogance and teach humility**

(Hindi: *Aur Vijay hame sikhata hai namrata. Aur mai is sadan ko viswas deta hun. Aur mujhe viswas hain yehaan ke jo hamaare seniors hain, kisi bhi dal ke kyon na ho, unke aashirwaad se hum us taakaath ko praapt karenge jo hame ahankaar se bachaye. Jo hamein har pal namrata sikhayein. Yahaan par (gesturing towards treasury benches) kitni hi sankhya kyon na ho lekin mujhe aap ke bina (gesturing towards opposition benches) aage nahi badna hain. Hamein sankhya ke bal par nahin chalna hain hamein saamuhikta ke bal par chalna hain.*) English: And victory teaches us humility. And I give this house an assurance - . And I have the faith that the seniors here, no matter which political party they are from, with their blessings we will acquire that strength which will save us from arrogance. That will teach us humility at every step. No matter how many numbers we have here (treasury benches - massive majority) I do not want to move ahead without you (opposition benches). We do not want to proceed on the strength of numbers but we want to proceed on collective/united strength.



*[I thank Prime Minister Modi of India and have presumed that the Prime Minister's office will not have any objections to me sharing the above extract(s) from their website in this book, whose ebook is freely downloadable/viewable by all, and whose paperback book is not-for-profit for me (author). This book does not have any financial profit motive whatsoever for the author.]*

## Education Related Parts of US President Obama's State of the Union 2014 Speech

Associated blog post date: 29th Jan. 2014, link: <http://eklavyasai.blogspot.com/2014/01/education-related-parts-of-us-president.html> , short link: <http://bit.ly/usprez-spch>

The above mentioned blog post provides extracts related to education from US president Obama's State of the Union 2014 address, <http://www.whitehouse.gov/the-press-office/2014/01/28/president-barack-obamas-state-union-address>, and some comments from me.

Here, in this book, I have used very limited quotes from above extracts of US President Obama's address. Comments of mine are prefixed by Ravi: and in italics.

- President Obama: "Today in America, a teacher spent extra time with a student who needed it, and did her part to lift America's graduation rate to its highest level in more than three decades." *[Ravi: This is the first sentence of his speech (after mentioning the persons the speech is addressed to). That shows how much importance the US president gives to teaching.]*
- Obama: "We know where to start: the best measure of opportunity is access to a good job." *[Ravi: I gave this quote as later on we see the importance given to education and training to make a person/student ready for that "good job".]*
- Obama talks about his administration creating high-tech manufacturing hubs and connecting businesses to research universities, all of which would help to create more high-tech manufacturing jobs. *[Ravi: The link between research universities and the high-tech economy is clearly recognized by the US president.]*
- Obama talks about a new manufacturing firm in Detroit which got a workforce from an American Job Center and which firm now has more than 700 employees. He talks of having asked Vice-President Biden to lead efforts to reform training programs in America so that Americans are trained with the skills needed by employers. He mentions connecting companies to community colleges so that the latter's training is designed to fill companies' specific needs. *[Ravi: Clear focus on providing specific training/education to meet jobs available in industry and the mention of community college in that context. I think India needs a good vocational training/education setup which offers a 2 year community college diploma/degree equivalent to students passing out of XIIth standard (which I think is equivalent to K-12 of the US). Can Information Technology education focusing on non-critical (IT) applications development be one area for such education/training? I don't know enough of vocational education in India to be able to answer the question but I think it is worth thinking about.]*
- Obama talks about teachers and principals across USA preparing students with skills for the new economy. *[Ravi: Full marks to the US government for trying to fix the problem. It would be so great if top Indian education policy makers also speak up publicly on such matters instead of it being a closed door and opaque license-raj education policy-making setup.]*
- Obama: "We're working to redesign high schools and partner them with colleges and employers that offer the real-world education and hands-on training that can lead directly to a job and career."

We're shaking up our system of higher education to give parents more information, and colleges more incentives to offer better value, so that no middle-class kid is priced out of a college education." [Ravi: Terrific!!!]

----

Ravi: It is so wonderful to see the top boss of a country, in a yearly stock-taking and future-direction address, give so much importance to education, and have a clear focus on education that imparts the skills the youth need in these job-wise challenging times. In contrast, in India, most of what I read about speeches made by top ministers and education policy makers is about research. I don't recall coming across any speech in recent times by any top Indian higher education policy maker about teaching Indian students useful skills that will help them get a decent job! (Note that previous articles in this book covering education related parts of Indian PM Modi's address and Indian President Mukherjee's address are of their June 2014 speeches which were *after* US President Obama's Jan. 2014 address covered in this article.)

BTW I saw the video of the speech as well. If you would like to do so here's the youtube video link I viewed, <https://www.youtube.com/watch?v=hed1nP9X7pI>, 1 hr. 29 min. 07 sec.

*[I thank US President Obama and have presumed that the US President's office will not have any objections to me sharing the above small extract(s) from President Obama's speech in this book, whose ebook is freely downloadable/viewable by all, and whose paperback book is not-for-profit for me (author). This book does not have any financial profit motive whatsoever for the author.]*

## USA - Student Right to Know Before You Go Act

Associated blog post date: 27th Dec. 2013, last updated on 5th Sept. 2014, link: <http://eklavyasai.blogspot.com/2013/12/usa-student-right-to-know-before-you-go.html> , short link: <http://bit.ly/right-to>

Update on Sept. 5th 2014: The act (or rather bill) mentioned below seems to have got stuck at the introduction stage. <https://www.govtrack.us/congress/bills/113/hr1937> states that it has only "4% chance of being enacted"! --- end update ---

I was very happy that some senators in the USA have introduced a new act which will empower USA students and parents with more information for making crucial higher education decisions.

Here's a press release dated May 9th 2013 about the act, "Sens. Warner, Rubio, Wyden Introduce Student Right to Know Before You Go Act", <http://www.warner.senate.gov/public/index.cfm/2013/5/sens-warner-rubio-wyden-introduce-student-right-to-know-before-you-go-act> [short link: <http://bit.ly/right-know> ].

Some important extracts from the press release are given below:

The legislation would streamline existing institutional reporting requirements to give students and their families more tools to easily compare graduation rates, student loan debt, employment prospects and potential future earnings as they make important decisions about higher education.

...

"Many high school seniors who are heading to college this fall have just paid their tuition deposits, and they likely have real questions about what value they are getting for their money," Sen. Warner said. "This

bipartisan legislation will combine relevant information in a rational way so that students and their families can access comparative information on which colleges and which majors will result in a good job.

...

“A college education is one of the most important investments an American can make in their lifetime, so it’s critical that we equip potential students and their families with as much information as possible,” Sen. Rubio said. “With this legislation we can finally provide meaningful, easily accessible data to make higher education decisions easier for the 21st century student.”

...

“There’s been a needed focus on access to higher education, but it’s time to bring value into the equation,” Sen. Wyden said. “Instead of forcing students to make blind decisions on such a huge investment, this bill would empower them with a wide range of information about what their choices will mean in working world.”

...

Rising educational costs and uncertainty in the job market have made the stakes higher than ever for individuals looking to invest in higher education. According to the College Board, the average cost of tuition and fees for the 2012–2013 school year was \$29,056 at private colleges, \$8,655 for state residents at public colleges, and \$21,706 for out-of-state residents attending public universities. Two-thirds of college seniors who graduated in 2011 had student loan debt, with an average of \$26,600 per borrower.

--- end extracts ---

This USA Today article, "Column: Learning blind", <http://usatoday30.usatoday.com/news/opinion/forum/story/2012-09-17/wyden-rubio-student-loans-college/57806404/1> [short link: <http://bit.ly/learning-b>] by Senators Wyden and Rubio, dated 19th September 2012, is about the same topic and very interesting (to me, at least).

Indian students \*must\* also have access to similar information about Indian institutions of higher education.

I am deeply thankful to USA senators Mark R. Warner (D-VA), Ron Wyden (D-OR) and Marco Rubio (R-FL) for bringing in this legislation. Readers may wonder why I should be thankful for this act/bill which has been introduced in the USA and not India. Well, I think this USA law can be a model for other countries in the world like India to emulate. I think it will be far easier for Indian legislators to bring in similar laws in India and convince Indian academic administrators and academics about its need, now that the USA has introduced it. [Feb. 2020 update: But it did not become law and I don’t know whether any other similar bill became law.]

*[I thank US Senators Mark R. Warner, Ron Wyden and Marco Rubio and have presumed that their offices will not have any objections to me sharing the above extract(s) from the above mentioned press release, in this book, whose ebook is freely downloadable/viewable by all, and whose paperback book is not-for-profit for me (author). This book does not have any financial profit motive whatsoever for the author.]*

----

Taking Charge: A State-Level Agenda for Higher Education Reform, <https://www.aei.org/research-products/report/taking-charge-a-state-level-agenda-for-higher-education-reform/> [short link: <http://bit.ly/31EUlZr>] dated July 2013 is an interesting view of US higher education and its challenges from a very top-level view. It mentions that, "There are also emerging doubts about the value of a college education. Although the cost of college has increased at three times the rate of inflation, little evidence exists that higher prices reflect higher quality." It refers to an Associated Press report which states that in 2012, 53% of recent college graduates were either unemployed or underemployed.

# First Question in 2nd US Presidential debate: Student asks about employment after graduation!

Associated blog post date: 19th Oct. 2012, link: <http://eklavyasai.blogspot.com/2012/10/first-question-in-2nd-us-presidential.html> , short link: <http://bit.ly/first-quest>

Yesterday I saw the 2nd US Presidential debate held on October 16th, 2012, <http://www.youtube.com/watch?v=QEpCrcMF5Ps>. Its transcript is available here: <http://www.nytimes.com/2012/10/16/us/politics/transcript-of-the-second-presidential-debate-in-hempstead-ny.html?pagewanted=all>.

The first question was by a young undergraduate student, "Mr. President, Governor Romney, as a 20-year-old college student, all I hear from professors, neighbors and others is that when I graduate, I will have little chance to get employment. Can — what can you say to reassure me, but more importantly my parents, that I will be able to sufficiently support myself after I graduate?" (Source: above transcript link from nytimes.com)

The transcript link also has the moderator saying that the questions were \*not\* known (prior to the debate) to the candidates and the commission (presidential debate commission, I guess) but known to the moderator and her team.

I think it is a celebration of US democracy that a young undergraduate student is able to pose such a real-life-concern question directly to the sitting President and his challenger. Hats off to US democracy!

The answers from the candidates were, well, politically correct :). But I am not sure whether they really were good answers. In my humble opinion, the challenges of today's tough economic times not only for the developed countries of the world but perhaps the whole world, must make educators/academics seriously examine whether the education they impart to students makes them job worthy. I think job-oriented education is the pressing need of the hour, worldwide.

A graduate who has been taught knowledge which cannot fetch him/her a job in today's economy undergoes huge suffering. The student's seriousness when he asks the question and how he clarifies that it is more important that his parents be reassured than him on whether he will be able to sufficiently support himself after graduation, says it all, IMHO.

# **Research vs. Teaching**

## **Nature and Science (Top International Research Magazines) 2011 Articles on Lack of Importance Given to Teaching Nowadays**

Associated blog post date: 18th Apr. 2013, link: <http://eklavyasai.blogspot.com/2013/04/nature-and-science-2011-articles-on.html> , short link: <http://bit.ly/nature-sc>

Here is a Jan. 2011 article, "Scientists Fault Universities as Favoring Research Over Teaching", <http://chronicle.com/article/Scientists-Fault-Universities/125944/>.

It refers an article in Nature, Jan. 2011, "University cuts show science is far from saved", <http://www.nature.com/news/2011/110112/full/469133a.html>, which states that funds for teaching have been cut in UK universities.

It also refers an article in Science, Jan. 2011, "Changing the Culture of Science Education at Research Universities", <http://www.sciencemag.org/content/331/6014/152.summary>, by 13 authors from 11 different US universities including MIT, Harvard and Yale, which states, "The reward systems at research universities heavily weight efforts of many professors toward research at the expense of teaching, particularly in disciplines supported extensively by extramural funding". It suggests seven initiatives for ensuring equal commitment of science faculty to their teaching and research missions.

Articles in Nature and Science, both of which are top scientific journals/publications, raising the lack of importance given to teaching, in 2011, clearly shows that research grant money is corrupting academic teaching ideals in the scientifically advanced Western world. Is India going to follow their path and make the same mistakes? Will young Indian students have to endure mediocre and poor teachers due to excessive importance being given to research in Indian universities? I think there is a serious risk of such things already happening and becoming more commonplace in future.

----

A comment on associated blog post:

Ravi S. Iyer wrote on April 19, 2013 at 4:52 PM:

I thought some readers may want to know that I sent a mail a few minutes ago with mostly similar content to the above blog post to appropriate Indian government ministers, top academic administrators, some NAAC executive committee members etc.

----

## **Suggestion of Separate Tracks for Teaching and Research by US Academics**

Associated blog post date: 30th Sept. 2012, last updated on: 2nd Oct. 2012, link: <http://eklavyasai.blogspot.com/2012/09/separation-of-teaching-and-research-in.html> , short link: <http://bit.ly/separation-of>

Is it time to have two separate tracks for teaching and research in academia (higher education)?

It seems that some faculty in the USA are talking about separation of teaching and research:

1) A tech. view from [Georgia Tech., one of the top US research universities in science and technology](#) here: [link broken in Feb. 2020] <http://www.gatech.edu/vision/big-ideas/separate-faculty-tracks-for-teaching-research> [Feb. 2020 update: Wayback Machine saved version: <https://web.archive.org/web/20130817084850/http://www.gatech.edu/vision/big-ideas/separate-faculty-tracks-for-teaching-research> (short link : <http://bit.ly/2SLshfh> )].

2) A humanities view from a [Columbia University Professor Emeritus, Herbert Gans](#) here: [http://www.huffingtonpost.com/herbert-gans/separate-research-and-tea\\_b\\_844575.html](http://www.huffingtonpost.com/herbert-gans/separate-research-and-tea_b_844575.html) [short link: <http://bit.ly/separate-r> ].

I could be wrong but I think that really seems to be the future for higher education in these times of economic uncertainty worldwide. The online education movement may push strongly for this separation.

In the case of CS and IT graduate/post-graduate degrees (e.g. B.Tech., M.Tech.) where, I presume, there is supposed to be substantial focus on teaching software design and development, excellence in teaching these topics should naturally go hand-in-hand with software contributions - if the teacher of these topics does not practice software engineering/development himself/herself how can he/she be an excellent teacher of software engineering? And what better measure of a teacher's excellence in the practice of software development/engineering than his/her software contribution record?

In my references above to software engineering I mean the actual practice of software design and development which includes, as a small part, study of various software development processes. Sometimes the software engineering subject is considered to be limited to study of software development processes which I think is a big mistake. Such limited view of the subject should be called software engineering process(es) and not software engineering.

## **India (and Other Emerging/Developing Countries) Should Not Obsess about Higher Education Rankings**

Associated blog post date: 17th April 2013, link: <http://eklavayasai.blogspot.com/2013/04/india-and-other-emergingdeveloping.html> , short link: <http://bit.ly/india-and>

Prof. Altbach, [http://en.wikipedia.org/wiki/Philip\\_Altbach](http://en.wikipedia.org/wiki/Philip_Altbach), wrote an article titled, "The overuse of rankings", <http://www.thehindu.com/opinion/op-ed/the-overuse-of-rankings/article4488869.ece>, in The Hindu dated March 9th 2013.

In the article Prof. Altbach mentions how top political and other leaders across the world, including India's Prime Minister, give tremendous value to university rankings like the Times Higher Education ranking (officially called World University Rankings), <http://www.timeshighereducation.co.uk/world-university-rankings/>, and the Shanghai ranking (officially called Academic Ranking of World Universities), <http://www.shanghairanking.com/>. The article states, "Prime Minister Manmohan Singh recently chastised Indian universities for having no institutions in the "top 200" of the global higher education rankings."

The author of the article digs deeper into the above two rankings. He states that the Shanghai rankings are research based (teaching quality is not measured). The Times Higher education ranking gives research a lot

of importance but also tries to factor in teaching quality and internationalization by using "weak proxies" to measure them. The article discusses some more issues related to the rankings.

In its conclusion the author states, "For India, or other developing countries, to obsess about the rankings is a mistake."

There was a response to this article from Mr. Phil Baty, editor, Times Higher Education rankings, "Why this global ranking process matters", <http://www.thehindu.com/opinion/op-ed/why-this-global-ranking-process-matters/article4603259.ece>, in The Hindu dated April 11, 2013. Mr. Baty warned that under-using the global rankings would be a bigger problem than overusing it.

Mr Baty wrote that Times Higher Education ranking (measures) "teaching, research, knowledge transfer and internationalisation".

Prof. Altbach wrote another article, seemingly in response to Mr. Baty's article, "Ranking obsessions and India's educational needs", <http://www.thehindu.com/opinion/op-ed/ranking-obsessions-and-indias-educational-needs/article4620733.ece>, in The Hindu dated April 16th 2013.

I am so glad to read the articles from Prof. Altbach, a distinguished educator, questioning the relevance of higher education rankings like Times Higher Education rankings or the Shanghai rankings for the vast majority of Indian universities/colleges (typically regulated by UGC/AICTE). While I am nowhere close to being a knowledgeable person on these matters from a country-wide perspective, it seems to me that his assessment that many of these higher educational institutions (of India) "mainly provide supervision of colleges and teaching in selected postgraduate fields, but perform little if any research" is correct. And, IMHO, there is nothing shameful about it! That's the role they are cut out to play, in terms of the finance available to them, the faculty available and the students they cater to. Teaching alone is not shameful but a very respected calling, IMHO.

I particularly liked him noting how eminent Indians castigate poor quality in Indian higher education. I think it has become a popular sport for India's top politicians. I would have hated to be in the shoes of the academics who had to listen to such castigation without any chance to respond.

Altbach wrote that a small number of competitive research universities is needed by India. I guess the elite IITs, IISc, TIFR etc. have government or other funds support and limited teaching load for its faculty, to shoot for these goals.

Altbach further wrote that overall quality of the higher education system has to be improved significantly, especially its colleges.

I think teaching quality needs a lot of improvement. BTW here is an interesting article about NPTEL tying up with some corporates (Google, TCS etc.) for its free online learning solutions: [http://articles.timesofindia.indiatimes.com/2013-04-12/news/38490818\\_1\\_nptel-technology-enhanced-learning-coursera](http://articles.timesofindia.indiatimes.com/2013-04-12/news/38490818_1_nptel-technology-enhanced-learning-coursera). They plan to have exams and certification for a fee (the teaching part of the course will be free). That may make it a very interesting option for many young Indians.

I think online education may 'revolutionize' Indian higher education landscape. I feel India is just about ready to get on to the MOOC bandwagon as the Internet has really caught on even in semi-urban India and some parts of rural India as well. If you want to read a short blogpost on why I feel MOOC may revolutionize higher education teaching, which is based on an external link article, you may please visit: <http://eklavyasai.blogspot.in/2012/11/napster-mp3-music-industry-disruption.html>.



# Improving Indian Academic Research and Teaching: Have Separate Research-Intensive Universities and Teaching-Intensive Universities

Associated blog post date: 7th February 2014, link: <http://eklavayasai.blogspot.com/2014/02/improving-indian-academic-research-have.html> , short link: <http://bit.ly/improving-in>

Today's The Hindu has an article titled, Paralysis in science policies, <http://www.thehindu.com/opinion/lead/paralysis-in-science-policies/article5661263.ece>. Its lead paragraph states, "Neglect of research in higher education has led to very low research intensity. Ninety per cent of our universities end up as teaching institutes where research is given a low priority for lack of funds"

A few days ago, The Hindu carried an article about the poor employability of Andhra Pradesh and Tamil Nadu engineering graduates, <http://www.thehindu.com/news/cities/Hyderabad/ap-fares-poorly-in-employability-of-engineers/article5639970.ece>. I think the article clearly shows that there is a serious teaching, and so graduate employability, crisis in engineering colleges in Andhra Pradesh and Tamil Nadu. I believe this crisis, to some extent, extends to other streams of education besides engineering, and to other states in India too.

What should universities do now - focus on improving teaching/learning outcomes and so improve employability of graduates and post-graduates (students doing masters' degrees) or focus on improving research? I think there is a conflict between teaching and research in many parts of Indian academia today.

One solution to this conflict/dilemma may be to have separate research-intensive universities and teaching-intensive universities. The teaching and non-research workload of academics in research-intensive universities must be low so that they get enough time to focus on research. The UGC/AICTE regulations for appointment and promotion of such academics must give high weightage to research output of such academics. [Please note that I am excluding the elite higher education institutions like IITs and IISc (and also medical education institutions).] Usually such research-intensive universities would be far more expensive to run than teaching-intensive universities.

The teaching-intensive universities would have high workload of teaching for its academics with some (less) time made available for research too. The UGC/AICTE regulations for appointment and promotion of such academics must give high weightage to learning outcomes for and employability of students taught by such academics. These kind of universities would be less expensive to run.

Of course, there should be migration paths for academics to move from one type of university to the other.

Today, I believe, we have only one set of UGC/AICTE regulations for appointment and promotion of academics of any type of university (research-intensive or teaching-intensive). That leads to situations where academics are denied promotion due to lack of suitable research publication output. Loading an academic with three to four courses of teaching load per semester and additional non-research work, and then upbraiding him/her for lack of good impact factor research publication output, and thereby denying him/her promotion, is not just being unfair, but being exploitative of the poor academic.

It also, unfortunately, in some institutions at least, creates an unhealthy environment for Masters students where they are expected/induced to choose project work that contributes to the academic's/department's research work even when UGC/AICTE regulations, I believe, permit a Masters student to do non-research project work. As an example, a software development project of suitable complexity and size is, I believe, permitted by AICTE regulations/norms (written/unwritten) to be considered as an M.Tech. Computer Science or Information Technology final year project. [Please note that many M.Tech. Computer Science or Information Technology students come from a different stream previous degree background like Production

Engg., Electrical Engg., Physics or Mathematics.] Such work would enhance software industry employability prospects of the student. But a research publication output obsession among academics may result in students being advised against doing such software development projects as their M.Tech. project and instead encouraged to do a research project that fits in the area of research done by academics in the department. The latter would contribute to academic research but may not necessarily be what the student needs and/or wants.

Having separate teaching-intensive and research-intensive universities who clearly inform students about their focus may help students choose the right type of university based on their needs and interests and their economic status. The lead paragraph of The Hindu article (Paralysis in science policies) looks down upon universities that "end up as teaching institutes". However, students, especially from poor rural and semi-urban areas, who are desperate for education that makes them employable may find such teaching-intensive universities to be a great blessing, and also find the tuition fees of such less-expensive-to-run universities more affordable. The academic research needs of the country can be primarily met by research-intensive universities who may be given the lion's share of tax payer money for academic research and who may then also be expected to deliver suitable results.

----

A comment from associated blog post is given below.

Ravi S. Iyer wrote on February 22, 2014 at 11:30 AM:

Here's a provocative article: Cut Off Harvard to Save America, <http://www.bloomberg.com/news/2014-02-19/cut-off-harvard-to-save-america.html>, by Richard Vedder. From [http://en.wikipedia.org/wiki/Richard\\_Vedder](http://en.wikipedia.org/wiki/Richard_Vedder): "Richard K. Vedder is an American economist, historian, author, columnist, and currently distinguished professor of economics emeritus at Ohio University."

The article mentions that top Ivy league US universities have large endowment funds. It goes on further to say, "Before endowments were large, professors sometimes had to earn their salaries by collecting tuition fees from students. When endowments provided professors a guaranteed salary, the incentive of offering high-quality instruction to paying students largely disappeared."

and

"A student graduating from Yale or Princeton, with their roughly \$2 million endowments per student, has a ticket to a well-paying job, while one graduating from the College of St. Joseph in Vermont, with its \$29,000 endowment per student, does not. Only 12 percent of the Yale and Princeton students have Pell's, compared with 71 percent at St. Joseph." [Ravi: Pell's refer to US Federal govt. Pell Grants. From [http://en.wikipedia.org/wiki/Pell\\_Grant](http://en.wikipedia.org/wiki/Pell_Grant): "Pell Grant is a post-secondary educational Federal grant sponsored by the U.S Department of Education. Enacted to help undergraduates of low-income families in receiving financial aid." The difference between elite and non-elite/commoner colleges in the USA seems to be stark. Perhaps if somebody does (or has done) a similar comparison in India, the difference found between elite and non-elite/commoner college education would be similar.]

Some additional info.

Richard Vedder is the director of the Center for College Affordability and Productivity, USA. Its mission statement: <http://centerforcollegeaffordability.org/about/mission-statement> says, "Founded in 2006, The Center for College Affordability and Productivity (CCAP) is dedicated to researching the rising costs and stagnant efficiency in higher education, with special emphasis on the United States. CCAP seeks to facilitate a broader dialogue on the issues and problems facing the institutions of higher education with the public, policy makers, and the higher education community. ..."

---

# Is Academic Research Grant Money Corrupting Academic Teaching Ideals?

Associated blog post date: 10th April 2013, link: <http://eklavayasai.blogspot.com/2013/04/is-academic-research-grant-money.html> , short link: <http://bit.ly/is-academic>

I think that is the case in many parts of Indian academia. However I do not have "hard data", as of now at least, to backup my view. Money or riches in some form has been an extremely powerful and even corrupting force since time immemorial. But money is also needed to run any show. The way money is spread in a system, slowly but surely, in a vast majority of cases, dictates how the people involved in the system and so the system, will behave over time. The parts of the system that are well funded will attract people and grow, and the parts that are starved of funds will repel people and decay. Academic research grant money has become a great magnet in academia while teaching seems to have become a poor cousin with very little or no grant money going to it. Naturally, most academics gravitate towards research grant money and look down upon "mere" teaching as a menial job.

This article on Forbes.com, written by a US medical science academic gives insight into research grant money influence in US medical science academia. It does not mention anything about a possible impact on teaching though, <http://www.forbes.com/sites/davidkroll/2013/01/05/top-10-reasons-being-a-university-professor-is-a-stressful-job/> [short link: <http://bit.ly/38whVqc> ].

Some notes on it:

2nd point of article: It mentions that some research universities assume that research project grants will cover some percentage of faculty salaries.

*[Ravi: My God! Faculty salaries being paid, even partially, by research funding seems to be a pretty scary thing to me. Naturally the focus will shift to getting research funding rather than excel in teaching, but then maybe teaching is not a big deal in a research university.]*

3rd point of the article: It mentions that research staff, fellows, editorial assistants etc. have to be paid from research grants. Once the grant is gone, all such staff are gone; the expertise is gone.

*[Ravi: This is horrifying. It is like running a business. I have experienced the stress of being a top manager in a start-up software company and worry about lack of business impacting ability to pay salaries of staff. But that I thought was part of start-up manager challenges. I did not realize that a tenured Professor could be having similar challenges that a start-up manager has. BTW both the tenured Prof. and the typical start-up manager would be capable fellows who can get another decent job but the staff working for them may find it difficult to get other decent jobs quickly, and that brings enormous emotional stress to the startup-manager (it had really stressed me out and affected my health with some effects being permanent) and, I guess, the tenured Prof.]*

4th point of the article: It mentions that administrators garner some decent percentage of research grant money for indirect costs.

*[Ravi: Oh Lord! So the administration chaps will use the extra money for their stuff (which may be very much needed for the institution). No wonder, university administrators, even in India, put immense pressure on faculty to do research. It is the money that research projects bring in, not only to a particular Lab. but also to the university administration that matters. Hmm. IMHO, there is a great danger of teaching ideals in universities being pushed somewhere to the corner by the pressure and power of research grant money. I have not heard of Indian government/UGC/AICTE giving teaching grant money - only research grant money. No wonder academics nowadays seem to be bothered mainly about research. I mean, if I were a regular/paid academic in a typical "research-intensive" academic institution that is what I would have to do*

Indian CS & IT Academic Reform (Past) Activism Blog Book Page 91

*too - focus on research grant projects and somehow manage the teaching stuff ensuring that there are not too many complaints.]*

--end notes on Forbes.com article--

In India, NAAC accreditation, <http://www.naac.gov.in/>, of universities & colleges has become a big thing, at least in the last decade or so, I think. Perhaps it is the only official way to differentiate between various UGC/AICTE universities in the country. I was surprised when I was informed some months ago by a recently passed out MBA student from a UGC/AICTE university, that its higher NAAC grading resulted in MBA recruits from that university being put in a higher pay bracket by some large company! So the HR guys of companies may be using NAAC ratings to decide fresher pay. I had also recently read a report of a state education minister using NAAC grading to differentiate between good and poor universities/colleges.

From what I understand of NAAC examination visits, the NAAC committee examiners focus on high impact factor journal publications. They are disdainful of conference publications! Of course, they look at various other criteria too. But high-impact factor publications is what really impresses them. I got the impression that they don't really give that much importance to quality of teaching - perhaps the problem there may be lack of availability of a standard, objective & official measure for quality of teaching like an impact factor for a research journal. But does that mean that a higher education official assessment and accreditation council can simply ignore teaching quality?!!! Perhaps Indian academia and NAAC can learn something from corporate training companies in India who focus only on teaching and, at least in the case of some famous software training institutes, have been an astonishing success over decades. They use student/participant feedback as one vital measure of teaching quality as viewed by the customer/student. As far as I know, such student feedback collection in Indian academia is quite rare, as of now.

I think NAAC introduces a fear factor for universities to engage in high-quality research that produces high-impact factor publications. Or else the university will not get a great NAAC rating/grade. So academic administrators desirous of having a good NAAC rating/grade have to push faculty to do high-quality research.

Another aspect of academic research is the "marketing hype" associated with large financial size research grant. Academics talk about the size of the research grant like how in the software consultancy industry, the financial size of an order is talked about. Larger the size, more the prestige and a large research grant is seen as a stamp of quality. I think the view is that bagging a large research grant is a reflection of the capability of the academics involved, as if they were not capable enough they would not have been given a large grant. I had checked with one or more international CS academics a few years ago, and they had concurred that large size of research grant is viewed as a kind of stamp of quality even abroad.

So senior faculty are actively "encouraged" to apply for research project grants. Getting a grant is celebrated like how bagging a project order is celebrated in industry! In very sad and marked contrast, teaching excellence is almost ignored! As teaching excellence does not bring in any large teaching grant money! That, I think, clearly shows the power of money to influence a system.

Okay, so what can be done to give teaching excellence its rightful share in Indian academia? IMHO, there should be substantial teaching grant money and further there should be an appropriate balance between teaching grant money and research grant money, given the type of educational institution involved (teaching-intensive or research-intensive). NAAC assessment/rating also should have a similar balance between measures of teaching quality and research quality.

-----

Given below are comments from associated blog post.

Ravi S. Iyer wrote on April 16, 2013 at 12:43 PM:

I can empathize with the anonymous gunning as well as petty vendettas in academia mentioned in point 1 of the Forbes.com article by David Kroll from some personal experiences that I have had on the suffering side of the anonymous gun and vendetta :).

----

Ravi S. Iyer wrote on April 17, 2013 at 12:44 PM:

I thought some readers may want to know that I mailed out the contents of the above blog post to appropriate Indian government ministers, top academic administrators, some NAAC executive committee members etc., on April 12th 2013. I have also included below the preamble I had in the mail.

I write this mail to you esteemed gentlemen out of concern about lack of importance given to teaching quality in UGC/AICTE regulated higher educational institutions in India, which are the vast majority of higher educational institutions in the country. The elitist IITs, IISc etc. are a special group which is very well funded by the Government of India - I am not referring to them at all in this mail. My concern is about the "commoner" higher educational institutions in the country. I must also add that my focus is on teaching quality in higher education which, in my humble opinion, is very different from quality of academic research in general (teaching/pedagogy research and related areas are an exception).

The views expressed in the mail below may be rather unusual. But they are the views that I have formed over a period of nearly a decade of free service of teaching software lab. courses and acting as a technical consultant for project work of M.Tech. (Computer Science) students in a deemed university in India, and also an Internet based study of the higher education world over the past two years or so. I am a software consultant and so an (ex) industry man - not an academic. My views are that of an outsider who has observed the system closely for around a decade. However, I may be wrong. So I welcome rebuttals and criticisms of this mail which may help me to correct my views, if they are wrong.

Some comments below (copy-pasted from a blog post of mine) have an informal language suitable for a blog post. Informal language sometimes helps in broaching sensitive matters without ruffling too many feathers. I request you esteemed gentlemen to please tolerate the informal language, at times.

--- Followed by above blog post contents ---

So far I have not received a substantive response from any of the addressees mentioned above.

----

## **CS & IT Academia: Is Teaching Excellence Important?**

Associated blog post date: 22nd Jan. 2012, last updated on 24th Jan. 2012, link:

<http://eklavyasai.blogspot.com/2012/01/cs-it-academia-is-teaching-excellence.html> , short link:  
<http://bit.ly/teach-excel>

At the outset I would like to say that I am not against CS/IT research. Without CS/IT industry & academic research, wonderful inventions of computer Operating Systems, computer Languages, Databases, the great & revolutionary force of the Internet etc. would not have happened. What I am against is teaching excellence in CS/IT academia suffering due to an unbridled enthusiasm for research.

In my considered opinion, "Teaching Excellence" seems to be not so important in Indian CS & IT academia. What really matters is "Research Excellence". The pressure to publish research papers and bag research projects of sizeable monetary value may be resulting in many teachers having no option but to dilute teaching ideals - they are also human and have to accomplish so many things in limited time. Of course, students have to be taught reasonably so that they do well at the exams but going beyond that to achieve excellence in teaching may be very challenging due to research pressure. Career advancement wise, there is no significant reward or recognition for "Teaching Excellence". "Research Excellence" gains recognition, even fame at times, and career advancement.

Why is that the case? An academic CS/IT department should focus on teaching, isn't it? Well, one look at UGC norms for appointment & promotion of academics, including CS/IT academics, gives the clear answer. UGC norms document: [19th Feb. 2020 update: [https://web.archive.org/web/20111120222535/http://www.ugc.ac.in/policy/revised\\_finalugcregulationfinal10.pdf](https://web.archive.org/web/20111120222535/http://www.ugc.ac.in/policy/revised_finalugcregulationfinal10.pdf) [short link: <http://bit.ly/39INmxu> ] shows a 20th Nov. 2011 version of following link which is now broken] [http://www.ugc.ac.in/policy/revised\\_finalugcregulationfinal10.pdf](http://www.ugc.ac.in/policy/revised_finalugcregulationfinal10.pdf). I presume AICTE norms will be similar.

"Teaching excellence" is not really a promotion criteria.

- For appointment as Assistant Professor a NET/SLET/SET qualified Masters' degree holder with 55 % marks is acceptable. But a PhD degree holder who is not NET/SLET/SET qualified can also be appointed as Assistant Professor.
- For promotion to Associate Professor, at least 3 publications for stage 4 Associate Professor, and 5 publications for stage 5 Associate Professor are needed (PhD is mandatory for direct appointment as Associate Professor).
- For promotion to full Professor, the norms are PhD + post-doctoral research output of high standard, patents & IPR, additional research degrees like D.Sc., D.Litt. Number of years of teaching experience also matter in the promotion to Associate Professor & Professor but it really is just the number of years.
- How well the person taught seems to be covered under Academic Performance Indicators - Category I: Teaching, Learning and Evaluation related activities, but they are based on the teacher's self-assessment! So, I guess, it will just be a formality with all teachers scoring well in it, irrespective of the reality :).

In other words, the unfortunate situation seems to be that, from UGC/AICTE norms point of view, Teaching Quality does not matter for career advancement, so long as students perform reasonably in paper based examinations. Very shockingly, for a practice oriented field like CS/IT, Teaching Quality for the practice oriented lab. courses also do not really matter and the unfortunate reality in most CS/IT departments in the country is that the lab. courses are notorious for being graded liberally so that almost everybody scores well!

Given the UGC norms for career advancement, most Indian CS/IT academics will focus on research instead of teaching. The CS/IT academic will, of course, try to do a decent job in teaching but it does not make sense for him to waste time trying to excel at teaching. He will get far better career rewards by being an excellent researcher who is a reasonable but not excellent teacher. Yes, of course, there will be CS/IT academics who excel at both teaching and research. The cream of the CS/IT academic field excel at both research and teaching. But I think, given the workload pressure, a significant number of CS/IT teachers would find it very difficult to excel both at teaching and research. You may also want to read another post of mine, "Is a PhD in CS/IT Necessarily a Good Teacher?" (article comes next in this book):

<http://eklavyasai.blogspot.com/2011/09/is-phd-in-csit-necessarily-good-teacher.html> [short link: <http://bit.ly/is-phd-in> ].

Some conscientious academics may choose to excel at teaching giving them lesser time for research and thereby not do well as a researcher. Their career growth gets stifled though they get the emotional reward of a clean conscience and earn the love and respect of students for being a good teacher.

Those teachers who choose to focus only on teaching and stay away from research will almost inevitably get into trouble with academic administrators. One cannot blame the administrators as they are bound by UGC/AICTE norms which, I guess, expect them to "encourage" research :). They also have to deal with the general expectation the leaders in government & industry have from academia to excel in research.

Such non-research oriented teachers may even be treated as liabilities even if they are excellent teachers as they, due to their disinterest in research, may not have a PhD. It may be as if the teacher is lowering the image of the department/college/university! Such persons tend to get fed up of Indian CS/IT academia and move into CS/IT software development industry (but not industry research) where their CS/IT knowledge/expertise is given a red carpet welcome. This is a sad state of affairs! In practice oriented streams like CS & IT such attitudes hurt rather than help. Who does it hurt? The poor students who lose the opportunity to gain from an excellent teacher who is disinterested in research.

Perhaps these UGC/AICTE norms work well for areas like Physics, Maths, Mechanical Engineering, Electrical Engineering etc. But for practice oriented CS/IT field I think they are very inappropriate. UGC/AICTE should treat CS/IT separately and come up with different norms where promotion is possible for excellent CS/IT teachers who are disinterested in research. Please note that UGC has separate norms for the practice oriented disciplines of Music & Dance, Drama and Visual (Fine) Arts. The CS/IT discipline has exploded into almost all facets of life all over the world in the past two or three decades with self-taught experts like Bill Gates (Microsoft), Steve Jobs (Apple) & Mark Zuckerberg (Facebook) becoming world famous icons. These three icons are/were technologists and not researchers. The fact that such self-taught experts have risen to very high stature in CS/IT industry seems to me to be good enough reason for UGC/AICTE to treat CS/IT differently.

I need to clarify that I am not making these remarks for the benefit of myself. By God's Grace, I retired from commercial international software industry over a decade ago. God's will/destiny led me to offer free service, mainly as a "visiting faculty", to Indian CS academia for teaching lab courses and being a consultant for project work for 9 years now. Given the strong practical know-how I have from over 18 years in the international software industry and the fact that I am offering free service, I have a plethora of opportunities for service of which teaching in CS/IT academia is just one possibility.

I am making these remarks in the context of regular (paid) CS/IT academics for whom the academic job is a livelihood. Imagine a CS/IT non-PhD but, say M.C.A. (Master of Computer Applications), qualified teacher who is also NET/SLET/SET qualified and who excels as a teacher but is disinterested in research. He will be a great boon to students. But, over time, as he sees that promotions are not coming his way, he will get dispirited and disinterested. Further, academia has a PhD club mentality and so he will be treated as a second-class citizen in academia. Either he will also see the "academic light" and start focusing on research relegating teaching excellence to the background OR he will quit academia.

Leaders from the Indian government and sometimes even from Indian industry talk about improving research in Indian universities. While that seems to be a very laudable goal, I think they should also talk about improving teaching standards in universities & colleges. In my opinion, a university/college is a portal first and foremost for teaching/learning (education) and then for research. But I am not really an "academic" and so maybe I am getting it wrong. I think I am an old school guy who feels that the FIRST and FOREMOST DUTY of a TEACHER is to TEACH and TEACH WELL.

# Is a PhD in CS/IT Necessarily a Good Teacher?

Associated blog post date: 30th Sept. 2011, last updated on 28th Oct. 2011, link:

<http://eklavyasai.blogspot.com/2011/09/is-phd-in-csit-necessarily-good-teacher.html> , short link:

<http://bit.ly/is-phd-in>

AICTE/UGC norms clearly imply that. And they are laying down guidelines for the entire country's higher education!!

In my non-researcher, non-PhD humble but quite logical thinker opinion there are broadly four possibilities:

- a) PhD qualified + Good teacher of CS/IT graduate/post-graduate students
- b) PhD qualified + Poor teacher of CS/IT graduate/post-graduate students
- c) Not PhD qualified + Good teacher of CS/IT graduate/post-graduate students
- d) Not PhD qualified + Poor teacher of CS/IT graduate/post-graduate students

AICTE/UGC CS/IT policy makers may say: How can that be? How can a PhD in CS/IT not be good at teaching CS/IT? That is impossible.

Further they might say, how can a non PhD be a good CS/IT teacher? That is not possible.

Well, I would not be surprised by such views from AICTE/UGC CS/IT policy makers. After all, they are the important members of the academic PhD club and have to protect the status & exclusivity of the academic PhD club.

But I think such views are FALSE. It doesn't matter if tons of CS/IT Professors think so. The TRUTH is not swayed by PhD arrogance. [Now, I am not saying all CS/IT PhDs are arrogant, but certainly some academic CS/IT PhD administrators are.]

Any CS/IT professional who has many years of experience under his belt knows the depth vs. breadth issue in software. Some guys may be focused on a narrow area and become in-depth in that area - like somebody who has worked for a decade only on compilers. Such a guy may be a veritable authority on compilers. But he may not know much about socket programming or n-tier architecture for E-commerce web applications - and he may be the first to state that it is not his area of expertise and that he does not know anything about it.

On the other hand some guys, especially software consultancy professionals, jump from technical area to area as they move from one customer project to another. So they may have an astounding breadth of expertise acquired over many years, say, in networking software products (gateways, LAN OS, browsers), E-commerce, mobile apps, Insurance software, Banking software, secure international financial transactions software, Object Oriented Analysis & Design, UML modeling, Design Patterns etc. This breadth enables them to take up software work in a variety of areas. However they may not be considered to be a veritable authority in any one area as they cannot match the expertise of a similar professional who has worked for the same number of years but in just one area.

The software industry needs both depth and breadth guys. In general, one could say that product development software companies need more depth expertise whereas software consultancy companies need breadth expertise. Of course, this is a sweeping generalization which will not hold true in all cases. But it may be valid for a significant number of professionals in both type of software companies.

The PhD academic in CS/IT is a very in-depth guy. Not only is he in-depth but is far more narrowly in-depth than a software industry in-depth guy typically. Further the PhD guy has an out-and-out research paper publication focus and not a software development focus. For the PhD researcher guy it is Publish or Perish.



But when the PhD has to teach CS/IT graduate/post-graduate students he has to be more like a software industry breadth guy. He may have to teach 3 to 4 courses ranging from Introductory Algorithms, Database Management, Compiler Construction to Object Oriented Analysis & Design! He could offer one course related to his research area - though there also he has to teach introductory or slightly advanced stuff and not his advanced research work. But he cannot limit his teaching to only his area of expertise - he has to teach other courses too.

So for the other courses, typically, he simply reads from the book, understands it and teaches it. Yes, many times he himself may have studied the course as a student. But then, as a student, you don't need to know the subject that well even to get good grades. When you teach you have to answer questions from students! There is no escape in the classroom/lab.! So he tries to master the book, or at least that part of it which he has to teach. It is book knowledge that is imparted to the students. What else can he do? He cannot do another PhD in the area which he is supposed to teach! Note that for the course he teaches where he has done research he will be very knowledgeable typically - I am talking about other courses here.

IMHO, such book knowledge teaching is nowhere close to the teaching of an industry professional who has practical expertise in that area. I repeat, nowhere close. And, just in case, readers are not aware, there is a lot of in-house teaching that happens in the software industry where knowledgeable seniors impart knowledge to juniors and peers in an informal or semi-formal teaching environment.

And then there is the communication skills issue. Research typically needs solitude and an individualistic approach. The researcher communicates on his research work with experts in his field. He can choose to, and many times, has to be very elitist in his communication - people who are not of the field may not understand him at all - which is OK as they are not from the field. In fact, a UK academician said part-jokingly but part-seriously that if others can easily understand your research work/talk then they think your research work is not that great! So only a few should understand, for your research work to be considered noteworthy :-).

However teaching requires very good communication skills and a non-elitist approach. What you teach should be understood by at least, say, 80 % of your class. You have to come down to the level of the students. Otherwise you may be a great researcher and very knowledgeable but a poor teacher. You may be a good research-guide as then you can be elitist but you may be a disaster as a teacher if you have an elitist approach in the classroom. By elitist here I mean that you speak in a way that only knowledgeable people of your field can understand you.

So while some PhDs may be good CS/IT graduate/post-graduate students' teachers, some may be poor teachers too.

And then you have non-PhD guys who are technologists and so know many tech. areas in reasonable depth but not in-depth. However the syllabus of most graduate/post-graduate CS/IT programs are not so in-depth. I mean, most of the text books followed are referred to by many industry guys as well. For example, Andrew Tanenbaum is famous for computer networks over decades in industry as well as academia. So many non-research but strong technologists from industry could easily teach many CS/IT courses in academia. Further they have practical on-the-job experience and so their teaching will be of a far higher-grade than a book-knowledge teacher. And most industry jobs require extensive team-work - so one's oral and written communication skills become quite strong. Further many teach short-term courses in industry.

So a non PhD can also be a good CS/IT graduate/post-graduate students' teacher. But being a non PhD he cannot contribute easily to any research efforts - further he could be a technologist who is utterly disinterested in research. And all non PhD software industry professionals are NOT good teachers - but some certainly are.

So, if AICTE/UGC policy makers think that insisting on PhD and research publications output is the way to improve quality of teaching CS/IT in academia I think they are making a TERRIBLE mistake. And, the

tragedy is that these policies may help the academicians PhD club to prosper but make life miserable for the poor defenceless students & parents of students who think that a CS/IT degree is a passport to an industry job.

But then they might say, who will do CS/IT research? How will the country progress in CS/IT research? Well, besides industry research, we can have academic research CS/IT departments - with degrees like M.S. by research. It becomes clear to students and parents that joining such degrees puts students on a research track. They will not expect to be trained to do software industry design & development jobs.

And these research CS/IT departments can be as elitist, as PhD clubbish as they want to be. Everybody will understand as they are research-centric. Such departments can be measured by the quality and quantity of research publications output (which seems to be the AICTE/UGC norm).

But most students and parents are not interested in research! They just want students to be trained to get a CS/IT job. So let there be academic teaching CS/IT departments which are teaching-centric and NOT research-centric. They should focus on providing a balance between theory and practical skills to students. When the student graduates he should be skilled enough to land a job on her/his own, without the help of campus interviews. That should be the acid test of whether an academic CS/IT teaching department is recognised by the Government (via AICTE/UGC) and NOT how many quality (impact factor based) research publications the department produced and how many PhDs the department produced.

Students at such teaching CS/IT departments may be given some limited exposure to research so that those who are intellectually capable and inclined (and do not have to immediately earn money from a job to get settled/help family or repay their student educational loans) can move into research in a research CS/IT department.

In India there is a huge amount of trust that innocent students and parents put on teaching institutions. Teaching institutions should be faithful to this trust by clearly informing students and parents right at the outset whether they are research-centric OR teaching-centric. Educational institutions should not attract students on the pretext of giving them job oriented skill sets but actually making them "theory & research oriented Generals" who struggle to get a "software development soldier" job (and don't get any "research General" job either).

CS/IT plays a vital part in the Indian economy now. It probably is the only area where India is recognised globally to be ahead of the relentlessly growing economic might of China. As of now Indian CS/IT professionals produced by AICTE/UGC governed academia are mainly 'theory & research generals' and some rare 'software development soldiers'. I do not know how Chinese CS/IT academia is. But given the Chinese penchant for being very practical and hands-on they perhaps produce CS/IT professionals with the right balance of 'theory & software development' skills besides a separate breed of PhD researchers. While US CS/IT academia certainly seems to have some similar problems (See "What Should We Teach New Software Developers? Why?" by Bjarne Stroustrup, [https://www.researchgate.net/publication/242499230\\_What\\_Should\\_We\\_Teach\\_New\\_Software\\_Developers\\_Why](https://www.researchgate.net/publication/242499230_What_Should_We_Teach_New_Software_Developers_Why) [short link: <http://bit.ly/what-teach> ]), it seems to give far more importance to code than Indian CS/IT academia. In the economic competition for software development business market share, it does not need a PhD to figure out that India will be left behind unless some corrective measures are introduced by AICTE/UGC CS/IT policy makers.

I am of the firm opinion that these AICTE/UGC CS/IT policies should not be left only to the academicians PhD club. Software industry experts must play a vital role in CS/IT academic policy making. After all, these issues are not of 'mere academic interest', the future of a key part of the Indian economy and the future of huge number of Indian youth are at stake.

# A Defense of The CS/IT PhD Teacher

Associated blog post date: 15th Oct. 2011, link: <http://eklavyasai.blogspot.com/2011/10/defense-of-csit-phd-teacher.html> , short link: <http://bit.ly/defense-of>

This is an edited mail exchange with a friend (Anakin probably) in response to the post: Is a PhD in CS/IT Necessarily a Good Teacher?

Friend wrote: First, let me be upfront with the fact that I may be in defensive mode - only because I have seen somethings very different here (West i.e. USA) and back in India, in some cases.

For pedantic, completeness reasons, I would like to add d) Non-PhD + Not a good teacher... We have both seen examples of this type, I am sure. They may not be immediately pertinent to the discussion at hand, but sometimes tend to negatively influence some policies.

Ravi S. Iyer (Eklavya Sai) wrote: Very valid point. I had missed it. Thanks. Added it to post.

Ravi S. Iyer had written in the post: And then there is the communication skills issue. Research typically needs solitude and an individualistic approach.

Friend wrote: I must disagree. While lots of R is individualistic, presentation skills are extremely important. Infact, every Graduate student is expected to go thru' comm. skills training, just like in the Industry.

Ravi S. Iyer wrote: Well, I think it is clearly recognised by researchers that presentation skills are important. I am not disputing that at all. But picking up presentation skills and good communication skills (spoken & written) is not easy for many. And so, quite a few PhD qualified CS/IT persons in India are not good communicators though they very much desire to be good communicators.

Friend wrote: Sadly, this is not the case in India, but in general, comm. skills are an absolute must.

Ravi S. Iyer wrote: I am really focusing on the Indian CS/IT academic system space. US is really different. My impression is that there it is a market driven system. I mean, tuition fees are so expensive that if a PhD qualified teacher is not a good communicator, students would crib and crib loudly - they would not care about whether he is a PhD or not. So a poor communicator PhD qualified teacher would not be able to last long as a teacher and may move to research-only positions.

Ravi S. Iyer had written in the post: So only a few should understand, for your research work to be considered noteworthy :-).

Friend wrote: Why is this the case? Because, only a few are capable of understanding this. Hence, it is all the more important to be able to effectively communicate to that elite few capable of understanding you. Consider the converse: if many could understand, you will probably gloss over most of the details and hope that those who can, will understand: this is definitely not effective communication, IMO.

Ravi S. Iyer wrote: The comment was made part-jokingly, part-seriously by a UK academician. I think many of the greats in sciences have the ability to convey their research ideas very effectively. I have been very impressed with how certain Western scientists present complex stuff. They can really communicate.

And I certainly recognise that very complex research stuff will be understood only by persons of the field. I mean I will not understand much about your research work, to be honest, as I am a technologist and not a researcher :-).

But I have noticed a common flaw in some persons of both the researcher and technologist communities. When they are not sure of their stuff they take refuge in jargon. And this jargon stuff well used can give an impression to not-so-knowledgeable people that the guy is knowledgeable!! If the guy gets grilled by a really knowledgeable guy who is not willing to get fobbed off by jargon & assumed-punditry, then his ignorance gets exposed.

That is why tech. reviews are so important in the software technology space. And I guess that must be the role that reviewers of scientific journals are supposed to play. The elite journals would surely be having quality reviewers but I think there are a lot of non-elite journals where high-sounding-but-low-real-content stuff passes through.

I think it is this aspect of the researcher that the UK academician part-jokingly referred to. I mean, the guy had a PhD under his belt and years of academic teaching experience in the West. It is not a non-PhD, non-researcher guy like me talking.

Ravi S. Iyer had written in the post: And all non PhD software industry professionals are NOT good teachers - but some certainly are.

Friend wrote: Isn't the converse also true? And therefore, only a few people are in general, really capable of teaching - not just restricted to the PhD circle.

Ravi S. Iyer wrote: I agree, all can't teach. Whether it is PhD circle or non-PhD circle. I presume this is what you mean.

Friend wrote: Why is this pertinent? Because, what I have seen is this: people who have spent more time with research, generally tend to know things that you can only know by that deep study. So, among the good teachers, those who have a PhD would tend to be the more knowledgeable ones and thus prepare better students.

Ravi S. Iyer wrote: They tend to know things in their research area well. And I have acknowledged that in the post. "Note that for the course he teaches where he has done research he will be very knowledgeable typically - I am talking about other courses here."

But for other courses? If a guy is a PhD in algorithms, will he automatically be knowledgeable enough to teach OOAD? IMHO, he won't. So he will read from a book, understand it well (as he has certain amount of intellectual capability - he did a PhD after all) and then teach. But that will be only book knowledge. How can it compete with a professional with years of OOAD expertise under his belt? Neither will it be able to compete with a PhD in OOAD.

And, as I wrote in the post, the teacher cannot do another PhD in OOAD just because he has to teach OOAD.

Now elite colleges & elite PhDs are perhaps special cases. In an IIT a teacher may offer only a couple of courses - one may be his research area and he can spend enough time on the other to know it reasonably well, even if it is book knowledge. And, anyway the teacher is a real smart guy.

But in commoner techie colleges in India for CS/IT I think the picture is very different. Bookish knowledge teachers are the norm rather than the exception. They may be teaching 4 to 8 courses in an academic year and only one CS/IT course may be corresponding to their research area.

I think you do not have exposure to the commoner techie college issues and commoner CS/IT teacher issues so much. I get the impression that your views are more appropriate for elite colleges and elite teachers, at least in the Indian CS/IT academia perspective. But, I could be wrong. After all, my exposure to Indian CS/IT academia is quite limited.

Ravi S. Iyer had written in the post: But most students and parents are not interested in research!

Friend wrote: Not sure about students, but afaik, many parents these days actively encourage their ward to go into Research/Teaching. Their reasons could be less stress, more respectable job (even if not pay ;) ) and quality family time.

Ravi S. Iyer wrote: Well, let's look at the numbers. I play it somewhat safe and say it is 80% industry job and 20 % research (what students do after passing out of grad/post-grad CS & IT degree programs). A noted CS academic and industry person wrote (in a mail exchange I had with him in early 2010) that he feels it is more 95 % industry job and 5 % research.

Ravi S. Iyer had written in the post: ...curricula not left to PhD alone...

Friend wrote: I can think of atleast one reason why this tends to be the case (I am by no means endorsing or opposing the statement). Education is as much about preparing people for the future as it is for the present. Not everyone is capable of picking-up skills; and so the system must consider the majority of the populace that cannot stand on its own. And, people can in a position to have some inkling about the future of science, technology, humanities, arts are those engrossed in R in those areas. Difficult choice - for both the people on curricula committees and for those deciding who gets to be on the committees as well.

Ravi S. Iyer wrote: Well, I think there should be transparency in the system. And the student should be able to choose what he wants to do. As far as I am concerned the Indian CS/IT academic system is heavily biased towards the elite. And that is why you have this new, "IT Finishing School" concept, which even IT graduates go to!!! Unbelievable. And, IMHO, a terrible proof of CS/IT academic system failure for the commoner students. For more please see: CS & IT Academia: IT Finishing Schools

But this is my view and I guess I get a little emotional about it. During my industry days, I have conducted countless tech. interviews and wondered what the he\*\* the teachers were teaching students doing programming courses in colleges. After spending some time in CS academia, I think now I know what the problems are when it comes to teaching programming in Indian CS & IT academia. Maybe that is clouding my vision about the bigger CS/IT academic picture.

## **Suggestion of Separate University Rankings for Research Excellence and Teaching Excellence**

Associated blog post date: 9th Dec. 2013, last updated on 13th Dec. 2013, link:

<http://eklavyasai.blogspot.com/2013/12/suggestion-of-separate-university.html> , short link:  
<http://bit.ly/suggestion-of>

I sent an email to Mr. Phil Baty of Times Higher Education Rankings, <http://www.timeshighereducation.co.uk/world-university-rankings>, with similar content to what is given below. The mail was copied to the editor of the Hindu.

I read your interview in The Hindu today, "Indian varsities lag behind in research", <http://www.thehindu.com/features/education/indian-varsities-lag-behind-in-research/article5433479.ece>.

While I believe the Times Higher Education Rankings may take into account teaching excellence in some way, in this interview you said, "The single best indicator in the rankings is the research impact, and here Peking performs pretty well — one of the strongest performances of any university."

In my humble opinion, such rankings as that of your esteemed institution may result in academic administrators and the academics who are administered by them, giving less importance to teaching and more to research.

The summary of a January 2011 article of Science magazine, "Changing the Culture of Science Education at Research Universities", <http://www.sciencemag.org/content/331/6014/152.summary>, has 13 authors from 11 different USA universities including MIT, Harvard and Yale, stating that "teaching load" is viewed as a derogatory label in STEM disciplines in many research universities, that "reward systems at research universities heavily weight efforts of many professors toward research at the expense of teaching" and that some institutions even reward research accomplishments and "raising outside research funds" by giving the concerned professors "teaching release"!

Please note that the above article is by 13 authors (academics presumably) affiliated to 11 different USA universities including MIT, Harvard and Yale.

Given this situation, I wonder whether your esteemed rankings institution can consider having two separate university rankings, one for research excellence and one for teaching excellence.

The teaching excellence ranking may be of great utility to the majority of students (and their parents) worldwide who go to (send their children to) universities primarily for an education and not necessarily research. BTW I have great appreciation for university research and am not against it in any way. But teaching should also be given its rightful place in a university.

--- end mail similar-content ---

I also forwarded the above mentioned mail to appropriate persons in Indian higher education.

-----  
Here's a short video, "Times Higher Education World University Rankings 2013-2014 results", <http://www.youtube.com/watch?v=FdRp7GHrcVU>, 6 min. 37 secs, giving a good overview of the importance and impact of these rankings. In the later part of the video the presenter, Mishal Husain (BBC news presenter), speaks to Phil Baty.

Some highlights of the video from my perspective (picked up using the transcript):

- \*) 2013-14 is the tenth year of these rankings.
- \*) Global research industry is bigger than ever; 7 million people doing academic research and 1000 billion US dollars spent on this research work.
- \*) The rankings are used by academics to decide on partnerships and career decisions; industry and philanthropists for investment decisions; 170 million people in higher education; students may be using it to choose where to study.
- \*) Top universities in these rankings are US and UK based with universities from European countries like Germany and Netherlands coming after them.
- \*) Asian universities from China, Singapore and South Korea are marching up the ranks.
- \*) Phil Baty says the reason for these Asian universities rising up is "pretty basic, I think it's about money".

\*) These countries (East Asian) have recognized the power of universities to drive the knowledge economy and so are investing money "at incredible levels" in their universities.

\*) In the West austerity has hit the universities and the effect of these austerities are seen in the rankings.

\*) The rankings have emerged as a geopolitical indicator. Success in the rankings is important to governments of some countries like India, Japan and Russia.

-----  
Update made on December 13th, 2013

I received the following mail response from Phil Baty of Times Higher Education Rankings:

Thanks for your suggestions – believe me that developing new and additional teaching indicators is high on our agenda.

I welcome your thoughts and your contribution to the discussions.

--- end mail response extract ---

I have given below relevant extracts of my response to him:

Thank you so much for your response.

It is very heartening for me to note that "developing new and additional teaching indicators is high" on your esteemed organization's agenda. I am also very encouraged by your acknowledging and responding to my mail.

In this context you may find the following two blog posts of mine to be of some interest:

a) Concrete Suggestions for Measuring Teaching Quality in Practice-Oriented Computer Science/Information Technology streams, [article is given earlier in this book]  
<http://eklavyasai.blogspot.in/2013/06/concrete-suggestions-for-measuring.html>.

b) Discussion on Concrete Suggestions for Measuring Teaching Quality ..., [article is given earlier in this book] <http://eklavyasai.blogspot.in/2013/06/discussion-on-concrete-suggestions-for.html>.

...

Thanks again and I wish Times Higher Education Rankings all the very best in its efforts to improve measurement of teaching excellence in its rankings.

--- end mail extracts ---

I would also like to mention that Phil Baty was fine with his response being shared/put up on this blog.

=====

A comment on blog post associated with above article, is given below.

Ravi S. Iyer wrote on December 11, 2013 at 3:56 PM:

A distinguished US academic to whom I had sent a mail with content similar to the post above, sent me this link, Peter Higgs: I wouldn't be productive enough for today's academic system,  
<https://www.theguardian.com/science/2013/dec/06/peter-higgs-boson-academic-system>.

I was stunned to read the article. I mean, a Nobel prize winner in science saying that he wouldn't be productive enough for today's academic system was just mind-blowing.

But, as I think about it, I guess the fundamental science work that he has done would need a very different environment from a paper-churning and somewhat noisy collaborative research environment which seems to be the norm for academic research environments today. [I really don't know much about his work but with my Physics graduate background I can get some idea of what an intellectual and perhaps imaginative feat it would have been to propose existence of a new elementary particle, which eventually got confirmed by experiments.]

I admire the honesty of Prof. Higgs that is clearly seen from this article. I find this honesty and truthfulness of leading scientists to be kind-of liberating. It is a privilege to read such honest words.

## CS & IT Academia: Research vs. Teaching

Associated blog post date: 22nd Sep. 2011, link: <http://eklavyasai.blogspot.com/2011/09/long-comment-8-mail-exchange-on-cs-it.html> , short link: <http://bit.ly/long-comm>

Note: This post is from a mail exchange related to the post+comments: CS & IT Academia: Serious Systemic Problems?, [article is given earlier in this book] <http://eklavyasai.blogspot.com/2011/09/cs-it-academia-serious-systemic.html> [short link: <http://bit.ly/serious-s> ].

Friend (Kartick Vaddadi) wrote:

I agree that teaching should be an important goal by itself, for academia.

I think you want only those people in teaching positions who actually know something first-hand. But that doesn't have to mean research. It could mean broad industry experience (you are an example here). I think the same bias we talked about earlier applies here, too. People who work in industry are considered to be less than researchers, so why would you want to dilute the department by letting them in? Hence you have the IITs saying that you need to have a PhD to be a prof, and you have to keep doing research. In fact, the IIT-Bombay CS dept promises profs that they have only one class to teach per semester so that they can concentrate on their research.

I (Ravi S. Iyer) responded:

Well, I just want at least some of the university CS & IT teachers to pursue excellence in design & coding and be rewarded for it even if they do no research.

Research will be the HOLY GRAIL of academicians including CS & IT academicians. But it should not be ONLY Research with NO DESIGN and little or NO coding.

About industry experience guys getting into academia - I think UGC/AICTE rules do allow for it provided the guy has a master's degree in that line (CS or IT). But perhaps at a lesser position than a 'researcher'. Not great, but at least they allow them to be teachers. My case is a real exception :-). B.Sc. (Phy.) [M.Sc. drop-out] teaching in a CS dept. as a regular university teacher?? No way. And I can understand that. The only exception allowed in such cases by AICTE/UGC rules is if the guy is an "eminent" person. So, I guess AICTE/UGC will not mind if some Indian university hires Bill Gates as a Professor, though they may crib if Mark Zuckerberg is hired even as an Assistant Professor - as they may not have heard of him :-).



I can be considered only as a "Visiting Faculty", which is exactly my status now. Suits me perfectly as I just am interested in Free Seva where I share my knowledge with students and am utterly disinterested in a regular 'academic' career.

IIT Profs. focusing on Research is somehow expected by policy makers, I think. I mean they expect IITs to figure in some top list of universities worldwide and I think these lists are driven primarily by research profile of the university. Must be quite a dilemma for an IIT CS Prof. - focus on country's technology problems using existing ideas or improvements on them OR do 'original' research that will be acclaimed internationally?

Friend wrote:

So, when universities say that profs should do or should have done research, I think the goal really is to weed out people who don't have any kind of original contribution or knowledge or experience and who just read aloud from the textbook. I think the goal is good, but it has been reinterpreted to mean research, which is bad. It's perhaps obvious when you think about it that the same bias that drives profs to concentrate on publishing papers also makes sure only those people get in the door to begin with.

Ravi S. Iyer (Eklavya Sai) had written in an earlier response: I have heard academicians being referred to as 'he has published 400 papers', which is supposed to mean that he is a great academic.

Friend wrote in response: That's actually scary.

Ravi S. Iyer had written in an earlier response: And this career growth measure naturally is a MASSIVE influence on the academic community. I mean, they are also human and would want more money and status.

Friend wrote in response: Agreed. And penalizing people financially for choosing to go into academia rather than industry is bad, if teaching is a primary goal of profs. If they are just publishing papers, fine, I don't really care whether they are paid less or more, but if they are teaching, and if our country has a bad shortage of talent that's holding back economic growth, then you shouldn't penalize people trying to fix this.

Ravi S. Iyer had written in an earlier response: A further, perhaps, very radical view is that some academic teachers should be given the option to be only teachers and not researchers.

Friend wrote in response: Agreed. I can imagine a university that lets people in with proven industry accomplishments, does not require them to have a PhD, and lets them focus 100% on teaching. Those people already have learnt a lot, and so are in a good position to teach others.

Ravi S. Iyer responded:

I am not primarily referring to industry guys here. I think a lot of teachers who are students turned teachers without any industry experience may actually enjoy imparting knowledge gained to students as a teacher. Teaching a subject forces you to learn it really well if you want to earn the respect of students, and almost any teacher worth his salt would want that. BUT the pressure to produce research publications directly eats into the time the teacher needs to master knowledge of design & programming. So most CS & IT teachers just manage with bare minimum knowledge of design and programming - anyway as they become senior they move out of Lab. courses and so they can be completely oblivious of design and programming from then on. I think that's the mindset - due to the research publication pressure.

Of course, there may be quite a few exceptions to what I have sketched above. But I think that is the norm for most CS & IT teachers. As Prof. Stroustrup mentions in his article some CS Profs. proudly say that "they don't code" - I guess they feel coding is meant for lower lifeforms :-).

Ravi S. Iyer had written in an earlier response: Sometimes I feel it is almost like a religious monk's quest for 'purity' - pure intellectual quest of a particular knowledge area untainted by application of that knowledge area to society.

Friend wrote in response: <http://www.paulgraham.com/philosophy.html>.

Ravi S. Iyer had written in an earlier response: I don't think I agree with your generalization that academia focuses on things that are original but ignores whether it is useful.

Friend wrote in response: I was oversimplifying. I didn't really mean that utility is 100% ignored, just that the goals are skewed. Isn't it true that academia rewards original ideas more than putting together existing ideas in a slightly different way or making subtle tweaks to something that already exists? But what if the best potential for improvement comes from an idea that already exists? You don't want to close your ideas to that and have an a priori assumption that you need a new idea.

After all, the whole point of ideas is that you can use them in umpteen situations. So statistically you're better off seeing if an existing idea can address whatever problem you're looking at, than go hunting for a new idea.

Ravi S. Iyer responded:

I have had very limited exposure to CS research paper publications. But, at least in the area of Web Services Security, I saw that some research papers did build on extending existing ideas in different ways. I think mature researchers probably value extending old ideas or perhaps even applying existing ideas to new problem areas. But overall my perception is that academic researchers tend to find new ideas more attractive than extending old ideas. Don't know whether industrial researchers look at it the same way.

From a technologist/software engineering perspective, application of well established ideas/approaches to problems demanding solutions is perfectly fine and can be quite challenging depending on the problem. New ideas delivering better solutions are also welcome. But what is important is the quality of the solution and the satisfaction it gives to the users and not whether the ideas/approaches used are old or new. That, I think, is a really key difference between a technologist's view and an 'academic' researcher's view.

Ravi S. Iyer had written in an earlier response: academic research does have a vital role in society.

Friend wrote in response: Sure, but couldn't it be even more if they get rid of their bias towards new ideas?

Ravi S. Iyer had written in an earlier response: I mean, if the student is smart enough to learn by himself why should he come to college to learn?

Friend wrote in response: To get a piece of paper called a degree certificate, so that Google will offer him a job :)

Ravi S. Iyer responded:

I think the Internet knowledge base is going to change all this. I mean a guy can learn so much from the net today itself. As online education portals become more organized and sophisticated I think they may become a powerful teaching alternative. To test whether a guy has learned or not they can have examinations - how the guy learned via regular brick-and-mortar universities or 'cloud campus' may be the choice of the student and/or parent.

Another major factor in favour of online education portals is the huge costs of university education especially abroad. I have heard horror stories of US students going into massive debt just to finish college and then they find they don't earn enough money to pay back the debt - they are into a debt trap BECAUSE OF COLLEGE STUDY. Horrifying stuff!!

Ravi S. Iyer had written in an earlier response: My experience is that it is quite easy to teach standard stuff to a smart student - the real challenge is to motivate and impart knowledge to the average student.

Friend wrote in response: Interesting. Yes, I don't have a teacher's perspective. Not being a teacher, I don't have to deal with the problem of the average student :)

---

A comment from blog post associated with above article is given below:

Eklavya Sai Maalik (Ravi S. Iyer) wrote on October 3, 2011 at 5:07 PM:

Long discussion spawned as a child post: CS & IT: Pure vs. Applied Research; Internet Based Learning, <http://eklavyasai.blogspot.com/2011/09/cs-it-long-comment-8-1.html> [short link: <http://bit.ly/cs-it-long> ].

## **Should M.Tech.(CS) Project be CS Research Oriented Or Software Engineering Project Oriented?**

Associated blog post date: 29th Apr. 2013, last updated 14th June 2013, link:

<http://eklavyasai.blogspot.com/2013/04/should-mtech-cs-project-be-cs-research.html> , short link:

<http://bit.ly/should-m>

The first part of this post has some views of mine which have had the benefit of an email conversation with a correspondent and discussion with a senior Indian CS academic. The second part of the post has an edited version of the email conversation with the correspondent which includes the views of the senior academic.

For readers who do not know much about the M.Tech.(CS) program I felt it appropriate to provide some background information.

### **What is M.Tech.(CS)**

An M.Tech.(CS) program stands for Master of Technology (Computer Science) and such programs are conducted in AICTE approved colleges/universities in India after getting approval from AICTE, <http://www.aicte-india.org/>, the key government constituted and supported technical higher education regulator of India. [Note this post does not focus on the elite technical education institutions of India, the IITs, [http://en.wikipedia.org/wiki/Indian\\_Institutes\\_of\\_Technology](http://en.wikipedia.org/wiki/Indian_Institutes_of_Technology), which also offer an M.Tech.(CS) or Indian CS & IT Academic Reform (Past) Activism Blog Book

(CSE) program. The IITs cater to a small percentage of technical education students in India. The vast majority of technical education students in India are catered to by AICTE regulated institutions and this post has a focus on M.Tech.(CS) or (CSE) offered by such institutions.]

The M.Tech.(CSE) program, I believe, is very similar, if not the same, as the M.Tech.(CS) program. It is just a matter of nomenclature with some institutions preferring the Computer Science & Engineering name instead of only Computer Science. So, I believe, all that is written in this post about M.Tech.(CS) applies to M.Tech.(CSE) as well.

### **Eligibility Criteria for M.Tech.(CS)**

I don't think the eligibility criteria for an M.Tech.(CS)/(CSE) is uniformly defined by AICTE. Different institutions seem to have slightly different criteria. Here is one from a Bangalore, Karnataka engineering college, which seems quite typical:

“Candidates who have acquired BE/B.Tech/AMIE or equivalent degree in Computer Science / Information Science / Electronics & Communication Engineering / Telecommunication Engineering / Electrical & Electronics Engineering by securing not less than 50% marks in aggregate are eligible (eligible). However in case of candidates belonging to SC/ST/Group, the aggregate percent of marks in qualifying examination should not be less than 45%. The admissions are through central counseling on the basis of merit in PG CET or GATE. A few seats are also available under sponsored and Management quota. GATE qualified candidates are eligible for scholarship through AICTE.”

Source: [24<sup>th</sup> April 2013 archived version of following broken link:

[https://web.archive.org/web/20130424113857/http://www.rvce.edu.in/mtech\\_cse.php](https://web.archive.org/web/20130424113857/http://www.rvce.edu.in/mtech_cse.php) [short link: <http://bit.ly/32iBi3O>] [http://www.rvce.edu.in/mtech\\_cse.php](http://www.rvce.edu.in/mtech_cse.php)

PGCET seems to be the state level Post Graduate Common Entrance Test for the state of Karnataka. Source: <http://kea.kar.nic.in/pgcet.htm>.

GATE stands for Graduate Aptitude Test in Engineering and "is an all India examination that primarily tests the comprehensive understanding of various undergraduate subjects in Engineering and Technology. GATE is conducted jointly by the Indian Institute of Science and seven IIT's (IIT Bombay, IIT Delhi, IIT Guwahati, IIT Kanpur, IIT Kharagpur, IIT Madras and IIT Roorkee) on behalf of the National Coordination Board – GATE, Department of Higher Education, Ministry of Human Resource Development (MHRD), Government of India." Source: [http://en.wikipedia.org/wiki/Graduate\\_Aptitude\\_Test\\_in\\_Engineering](http://en.wikipedia.org/wiki/Graduate_Aptitude_Test_in_Engineering).

SC/ST group refers to SC/ST reservation, [http://en.wikipedia.org/wiki/Reservation\\_in\\_India](http://en.wikipedia.org/wiki/Reservation_in_India), which in the education system, provides for lower thresholds of entry and some other benefits to certain categories and castes of Indians who historically have been deprived/depressed.

However there are institutions that allow M.Sc. i.e. Master of Science Maths/Physics students to also pursue M.Tech.(CS). An example is a university in the state of Himachal Pradesh, <http://www.mmusolan.org/courses.php>:

“The candidates should have passed with minimum 50% marks (45% in the case of reserved category) in aggregate in B.Tech/B.E. in Computer Engg./Computer Sc. & Engg./ Computer Technology/Information Technology/ Electronics Engg/Electrical Engg. or M.Sc. in Computer Science/ Electronics/IT/ Software Engg./ Maths/ Physics/Statistics or MCA or its equivalent examination from any recognized University.”

Even the elite IITs allow M.Sc.(Maths) students, who qualify and get a good rank in the GATE (CS) exam, to do an M.Tech.(CSE) as I know of such a case less than a decade ago.

## **M.Tech.(CS) Project**

I could not locate suitable links on the net describing the academic regulations for the M.Tech.(CS) project. From experience I can say that it is typically referred to as the final year project (2 semesters, namely third and fourth semesters of the 2 year M.Tech. program). The third semester may have some additional courses to be done but the fourth and final semester is exclusively for the project work. At one institution, 10 credits are associated with the project and I believe that to be quite typical.

The evaluation components of the project, from my experience, are a large component of external examiner evaluated thesis, smaller component of project viva and perhaps some other small components evaluated by project supervisor/guide. I believe this evaluation pattern to be a typical one for AICTE regulated educational institutions.

## **M.Tech.(CS) Project Types**

From my experience, broadly there are two types of projects, software engineering project oriented where the student gets exposure to problem requirements, design, development and test for a significantly sized and complex system software type problem in CS areas like web services security, image processing, parallel processing, ontology (for primary & secondary school education - roughly equivalent to K-12 in USA, I guess), object oriented analysis and design etc. I refer to this type as software engineering type. Such type of projects typically have no research component. They may involve studying associated computer science literature but will typically not involve publishing a research paper on the project.

Then there is a research oriented type of project where the student first studies computer science research literature for a research problem suggested by his/her supervisor and then works on the research problem. Typically there is some software engineering work as implementation of approach adopted to solve the research problem. During the course of the project or after the end of the project, the student in conjunction with his/her supervisor may also publish a research paper or two from the project work - typically conference paper(s).

**[Jargon Confusion:** Software engineering is a research area as well. In this post I am not addressing that area of research at all. In Indian CS & IT academia, the term software engineering research seems to be used quite rarely as compared to computer science research. In this post I am differentiating between software engineering research and software engineering software development work by referring to the latter as 'software engineering project' or simply 'software engineering' without the word research.]

## **Should M.Tech.(CS) Project be CS Research Oriented Or Software Engineering Project Oriented?**

As the M.Tech.(CS) program usually allows students with Bachelor's degree in non-CS areas to enter the program, many students entering the M.Tech.(CS) program may not have done a software engineering project as part of their Bachelor's degree. Further they may have basic programming skills like using C++/Java for some kind of work but not be really proficient in using them for complex system software kind of tasks. E.g. Using C++/Java/equivalent programming language along with, say, an Image processing class library may be something which may severely test an M.Tech. student who is mediocre in programming.

In my humble opinion (IMHO), before advising an M.Tech. student to take up a particular type of project, the supervisor should assess the student's capability in software engineering project work involving requirements study, design, development and test of a fair size problem of system software type complexity (as against simple Database forms oriented applications). If the student is weak in software engineering project work then the student should be advised to do a software engineering type project and there is nothing wrong about it as, I am given to understand, AICTE regulations permit such projects with zero research component to be considered as M.Tech. projects. [I tried to locate/download AICTE (model) regulations for M.Tech. projects but could not find it.]

If, on the other hand, the student is found capable/experienced enough in software engineering project work of system software type by the supervisor, then the supervisor can, or perhaps should, advise the student to take up a more challenging computer science (or software engineering) research oriented project (which will also typically have some software engineering project component). This allows good and excellent students to get the best out of their M.Tech. project - research angle exposure as well as some software engineering project work.

But "actively encouraging" most or all M.Tech. students to take up research oriented project work in AICTE type of universities (unlike the elite IITs) can be ill-advised if the regulations allow for zero research software engineering projects. IMHO, it is an unethical practice to "actively encourage" M.Tech. students to do research projects (or scare them by veiled or open threats of poor marks for zero research component software engineering type projects) when the academic regulations allow for zero research component software engineering projects.

If a department wants to have a Masters program which is only research oriented there are AICTE approved programs like M.S. by Research which a department can take up. In such cases there will be no confusion from regulations point of view at all.

Another point is about rural university/college issues. Typically such universities/colleges have significant number of rural students. Many of these rural students who enroll into an M.Tech.(CS) program may not really be strong enough technically to take up research work in their M.Tech.(CS) program. Expecting all such students to do research as part of the M.Tech. project work and then getting disappointed by the work they do seems to be a grave error of advice/judgement.

Yet another point is the need for senior academics to have research assistants for doing research work and publishing research papers. In places where senior academics do not have enough PhD scholars (PhD students/grad-students) or M.S. by research students associated with them, the senior academics may view the M.Tech.(CS) students as their research assistants! That, IMHO, is not an ethical practice.

IMHO, academics must always bear in mind that their first and foremost duty towards M.Tech. students is the teaching part of their mission; the research part is secondary and optional. Most M.Tech.(CS) students look for a software engineering job (systems software type usually) in the industry after passing out. If they have not had good exposure to a software engineering project of system software type, they may either struggle to find a suitable job or worse, get a job and then struggle to perform adequately in the job. IMHO, M.Tech.(CS) post-graduates passing out through the educational system without decent software engineering project exposure of system software type is a failure of the teaching mission of Indian CS academia.

---

This second part of this post has an edited version of the email conversation with the correspondent (who is currently based in USA) which includes the views of the senior Indian CS academic. It is rather long and so please feel free to skip reading it or just skim through it.

Ravi wrote:

In the blog post, "Nature and Science 2011 Articles on Lack of Importance Given to Teaching Nowadays", (article included in earlier part of this book) <http://eklavyasai.blogspot.com/2013/04/nature-and-science-2011-articles-on.html> [short link: <http://bit.ly/nature-sc> ], I plan to add the comment below:

In some extreme cases, research glory/publication count obsessed administrators may even create a culture where Masters students are "actively encouraged" to work on research projects thereby contributing to research output of the department/institution but many times at the expense of their Masters learning goals. The poor students may not even realize where they are losing out while they are doing their Masters. When they start looking for non-research jobs like applied science and technology jobs after their education then

they will realize the shortcomings in their education. [Industry and academic research jobs, I am given to understand, are more suited for PhDs. A non-PhD person joining a research job may find a PhD glass ceiling staring him/her in the face very quickly in their career.]

A PhD scholar/grad-student (graduate doing a PhD) working on research is a completely different matter, of course. Same is the case with an M.S. by research student. That is what s/he is expected to do. I am talking about Masters students like M.Sc. and M.Tech. students. Encouraging the typical M.Tech. student to publish a conference paper while s/he is doing their project work is, IMHO, a very dangerous and unethical practice. While the student is doing an M.Tech., s/he should focus on learning the technologies involved in his/her project and aim at excelling in that area. Once the M.Tech. project period is over, including submission of project thesis (for evaluation), and the viva/exams, s/he exploring the possibility of publishing a conference (or even journal) paper based on their project, seems quite fair and ethical to me.

Some exceptional M.Tech. students may be brilliant who not only can publish some research papers (conference papers typically) but also learn the technology skills part very well. But most M.Tech. students will lose out on picking up technology skills well if they are "encouraged" by their supervisors/guides to publish a research paper while they are in the middle of their M.Tech. project work. [A classic case may be a project supervisor "encouraging" an M.Tech.(CS) student to try out a research approach using MATLAB instead of C++/Java as the student is not able to write such C++/Java programs quickly. The student may pick up MATLAB skills but may end up keeping his/her C++/Java/(other well known programming language) programming skills applied to a particular domain like Ontology (Automatic Classification/Automatic Annotation) or Image Processing, by using associated class libraries, which are the vital skills an M.Tech.(CS) student is expected to be proficient in, at a lower level than the desired level!]

Such practices will help the supervisor/guide (as they will be co-authors of the paper) and it will help the department/institution, as it will increase the publication count but, IMHO, it is an unethical practice, as most M.Tech. students will lose out on the learning technology part. I think such a practice, if followed, will be a clear case of academic research craze corrupting academic teaching ideals.

[Ravi: As the comment above is now part of this post, there is no need to add it as a comment to the other post referenced above.]

---

Correspondent (Corr) wrote: I've been mulling over what I should say for some time now. First, I am not sure I am an appropriately unbiased person in this matter: I've been fortunate to have a reasonable balance between the CS and programming; notice that I am distinguishing the two. I have seen far too many botched projects because of a lack of understanding of basic CS concepts. All I contend is that you need both: programming skills and sound CS fundamentals.

IMO, what happens when people don't understand basic CS is what I'd like to call the copy-paste programmer. He may be good enough to write assignment programs, but not really industry projects. He will use the miserably slow linear scan-and-search when he could be using something much better. And so on.

Ravi wrote: For system software projects, I agree with your view. Fundamental theory of CS (as against advanced stuff which I consider as optional for some type of system software projects) + programming fluency makes the ideal combo. Only programming fluency without fundamental theory of CS makes for a weak systems programmer typically. [As a mostly self-taught and mostly systems software programmer in the mid 80s to around the end of the millennium, I had to struggle with my theory gaps by doing some intensive self-learning of some CS topics. I was not the ideal systems programmer due to lack of a proper, formal CS theory background but the supply of knowledgeable CS theory + programming guys (as against formally CS qualified but without good knowledge of CS + programming) was so low that people like me with intense self-learning periods, had to do such jobs and did it to quite some satisfaction of the customers usually. In today's context I really do not know whether significant number of CS graduates and post-graduates in India are knowledgeable enough of fundamental CS theory. Of course, there will be more numbers of good

CS graduates now than my times (1984 to 2002). However, the demand for system software chaps must have gone up exponentially in recent times in India. So don't know whether the good CS graduate numbers have grown in a matching exponential way or better.]

Corr: Coming back to the problem addressed in your comments: departments at universities usually pitch themselves as Research-oriented or Teaching-oriented; or at least what the majority of the faculty think they do is pretty clear. I think there is sufficient onus on the student to do his homework and find out what he is up for before admission. Once a student joins a research-oriented department, I am not sure who is to blame. What can the supervisor do if the student is not good enough to program his project in C/C++/Java? Should he allow the student to lag and potentially fail his degree? Matlab like environments are an easy compromise. Moreover, IMO, the masters projects are hardly a place to learn programming; rather, that is where students prove that they can effectively apply what they learnt: both in terms of programming and CS basics. I see the Masters projects as a place to fine tune and polish existing skills more than a place to 'learn' programming.

Again, please remember that I have a natural bias in this regard.

Ravi: And I may have a bias too. So let us have a free exchange of our views knowing that either or both of us may be rather wrong. The exchange of views can help in possibly both our views becoming more nuanced.

About M.Tech. degree: Initially when I was in industry I thought M.Tech.(CS) chaps would be really great guys technically as I sort-of presumed that it will be a B.Tech.(CS) + M.Tech.(CS) combo. When I checked around somewhat recently I was told that even at IIT usually a B.Tech.(CS) guy does not do an M.Tech.(CS)! It is guys from other streams (Other engineering streams, Maths, Physics etc.) who want to change their stream and get into the CS stream that do M.Tech.(CS).

So the typical M.Tech.(CS) student may not have done a strong software engineering (req. + design + programming + test) project prior to his/her M.Tech.

Certainly an M.Tech.(CS) project cannot be a programming assignment. It has to have a project flavour as against a programming assignment flavour. And what is a project flavor: the problem should be of significant size, should involve req. analysis, design, development and test - all of this work should be appropriate for a 10 credit course (in my experience at an educational institution that is what it was IFIRC; at other places it may be slightly different). The project report should present all the work done so that it convinces the examiner that 10 credit equivalent amount of work has been done and that it is of sufficient technical complexity (system software type) suitable for an M.Tech. degree.

Based on interactions with experienced Indian CS academics I was told that this is how an M.Tech. project is viewed elsewhere in Indian academia too.

About departments pitching themselves as research-oriented or teaching-oriented, I think the crux is the academic regulations for a particular degree program. If the regulations for it (e.g. M.Tech.(CS)) provide a zero research software engineering type of project to be taken up as an M.Tech. project then the department must allow a student that option.

*You wrote, "the masters projects are hardly a place to learn programming; rather, that is where students prove that they can effectively apply what they learnt: both in terms of programming and CS basics."*

I agree in general. I repeat that the Masters project cannot be a programming assignment. It has to be a full project. And a full project will typically involve CS basics and programming.



*You wrote, "What can the supervisor do if the student is not good enough to program his project in C/C++/Java? Should he allow the student to lag and potentially fail his degree? Matlab like environments are an easy compromise."*

I think this is a vital part of our discussion for me. I think your view is what some CS academics in India take. Maybe you and they are right. But I think the CS academics are terribly wrong from a teaching point of view. Maybe I am wrong. Anyway, here's my view:

An M.Tech.(CS) student who cannot do a technical project in C++/Java/equivalent programming language but slips in something in MATLAB, should not be given a CS degree, IMHO. He/she does not have the capability to provide system software engineering solutions. He/she should be forced to learn system software engineering skills (req. + design + programming + test) before he/she is awarded the M.Tech.(CS) degree.

Matlab environments are an easy compromise only from research results point of view. But in reality it is the poor M.Tech.(CS) student who suffers by not getting a chance to learn good systems programming as part of his/her project work (in the case where the student is weak in such programming skills).

I feel people who are weak in software engineering should not be allowed to take up research oriented M.Tech.(CS) projects. If a supervisor finds that the student is not able to implement some paper in C++/Java quickly, the supervisor should change the M.Tech.(CS) project of the student to a software engineering type instead of research type in the interest of the student and in the interest of academic honesty & integrity. And such a project is in full conformance with AICTE regulations - that is the vital thing in AICTE regulated institutions/departments in India.

An M.S. by research in India gets evaluated differently I am told. That seems understandable to me. So a guy doing his project work in MATLAB for an M.S.(CS) by research may be fine. But not an M.Tech.(CS).

---

Corr: 1. Let me clarify further on the copy-paste programmer. Some persons earn a name for being capable of beating code until it "works" but they may not have a good CS background. They use copy-pasting from other sources/web-sites without having an understanding of how that code works. When instructed to implement concept X/algorithm Y (which is a common way that requirements are given out) they cannot understand basic things about the algorithm and cannot implement it. So they try to use copy-paste techniques without understanding the copy-pasted code. While such cases may be chronic cases, I say this just to point out that it is not necessarily true of only/mostly system projects needing basic understanding of CS fundamentals.

Ravi: In my generation/group of self-taught systems software programming guys I think a programmer who could not explain code that he copy-pasted from somewhere else into his own 'production'/released program would be considered an unreliable and bad programmer.

Corr: 2. Regarding the M. Tech degree: I don't think it is true in general that mostly non-CS people do M.Tech. It is definitely the case in some institutions, but not in general. Specifically, IIT-ians don't bother with M. Tech mostly because their B. Tech degrees are good enough to get extremely high-paying jobs. Trend was, when I graduated, most IIT-ians end up doing MBAs from IIMs. That is the trend with IITs. If you look at the admission process to the IIT-Bombay M. Tech (CS) degree, you need to clear the GATE-CS exam (<http://www.cse.iitb.ac.in/page33>, see the 'CSE Department Admission Requirements and Procedures' section).

Ravi: IIT + IIM (IIMs, [http://en.wikipedia.org/wiki/Indian\\_Institutes\\_of\\_Management](http://en.wikipedia.org/wiki/Indian_Institutes_of_Management), are the elite management education institutions of India) was the prized combo even in mid 1980s when I started Indian CS & IT Academic Reform (Past) Activism Blog Book

my career. Most top software consultancy companies then including Datamatics, <http://www.datamatics.com/>, where I was employed (considered then, I believe, to be 3rd/4th top software consultancy company in India after TCS, TBL - Tata Burroughs Ltd, and sometimes after PCS - Patni consultants ... - Infosys was not born yet – Mr. Narayana Murthy was working with PCS then I believe) recruited some of such "stars". And some such stars were pretty good, I must say. I was very impressed with one such person with whom I had a professional and personal friendly relationship for many years. He became a director in a medium sized international software consultancy company.

Had a discussion with a senior Indian academic today on matters related to this mail exchange. His view was that B.Tech.(CS) or B.E.(CS) rarely do M.Tech./M.E.(CS) from same college/university. But many who do B.Tech.(CS) or B.E.(CS) from mediocre (and poor, I guess) universities/colleges pursue M.Tech./M.E.(CS) from better universities/colleges. That seems to make sense to me.

Regarding IIT M.Tech.(CS) needing GATE in CS clearance. Yes, I guess. But even GATE would not be able to test software engineering project kind of exposure of the student well. So you can certainly have many cases of people entering an IIT M.Tech.(CS) program with zero software engineering project experience.

In this connection I would like to share some personal experience. Around 1990 or 1991, as a software manager of a startup firm in SEEPZ, Mumbai, <http://en.wikipedia.org/wiki/SEEPZ>, I recruited an M.Tech.(CS) IIT Bombay passed out student. I recall that during the interview I conducted with him, he was talking about a radar technology project. I did not understand the physics/electronics part but I did try to dig in the software design & development part of the project. He could not give me a good account. I thought maybe I did not do the interview well - after all he is an M.Tech.(CS) from IIT Mumbai. And he did explain some programming stuff quite OK (separate from project). I gave the okay for recruitment. Later I saw he was a little slow on software development initially. But he picked up the stuff quickly - so there was no eventual problem.

I also learned that he had done B.E./B.Tech.(Production Engg.) and then moved into M.Tech.(CS). That seems to explain his poor software engineering project experience - zero software engineering project experience in B.E. and research type project in M.Tech.(CS) at IIT Bombay.

Corr: M. Tech, by virtue of its name is a technology program. But it is not uncommon that the projects be research oriented. For example, CSE IIT-B clarifies this upfront: <http://www.cse.iitb.ac.in/page31> (Postgraduate Programme (M. Tech.) section, first para, last line). The way I see it, this is a symbiotic approach. Universities mostly thrive on research. While teaching helps the students, research helps the university. And, mostly no sane teacher/lecturer/professor asks for research from Undergrads. But the moment one is enrolled in a masters program, I suppose it mostly requires some research. How else would you show a value-add for the masters program?

Ravi: Hmm. My non-academic, man-on-the-street view is that a Masters degree can involve teaching of advanced topics not taught in a Bachelors degree. That is the value addition. The research component may be optional depending on the kind of degree offered. But then I do not know whether this view matches the academic view. And that view is more important here.

Corr: I think if programming is to be taught, it should be at the undergraduate level.

Ravi: I don't disagree. And I think that is the case in most academic setups in India too. However they may not teach stuff like Advanced Unix programming, Socket programming, in a B.E.(Civil) program or an M.Sc.(Maths/Physics) program.

Corr: Even in the US universities, programming capability is assumed for the Masters program. If you cannot program, the professor can atmost direct you to enroll in an undergraduate programming class and quit the current masters class. Masters programming oriented classes mostly focus on things like language

Indian CS & IT Academic Reform (Past) Activism Blog Book Page 114

support for certain design patterns, C++11 features etc. And mostly such classes are offered as single credit, seminar electives.

I think a (possibly better) approach would be to have more strict filtering during the admissions, rather than failing the student for lack of programming skills. To me, failing a student for lack of programming skills is simply an acknowledgement of the failure of the admission process whose goal was to filter and choose sufficiently prepared students.

Ravi: Well, there are various types of programming skills. The entrant to the M.Tech. program may have basic programming skills like using C++/Java for some kind of work. But using C++/Java along with an Image processing class library may be something which may test a mediocre in programming student severely. And that's where him using the MATLAB route for quicker research results, IMHO, is a terrible option from an M.Tech.(CS) technology learning perspective.

About admission process being weak: I think the practical reality in India is that many M.Tech.(CS) students enter the program with poor software engineering project work skills. You have to deal with such students.

Corr: 3. Finally, there is the humanitarian aspect to education. Barring a few statistical exceptions, many supervisors sometimes want to overlook the students' shortcomings and hopefully help or push them through the program. They need to find such methods to help them out.

4. I definitely agree with this though: I feel people who are weak in software engineering should not be allowed to take up research oriented M.Tech(CS) projects. However, I think supervisors must allow leeway for the student to experiment with their own capabilities. But yes, the supervisor should be proactive to point out and change the direction of the project if needed.

Ravi: That (point 4 above) is the crux of the whole discussion for me. I took the opinion of the senior academic I mentioned earlier, on it. We agreed that the supervisor should assess the student's capability in software engineering project work. If the student is weak then the student should be advised to do a software engineering type project and there is nothing wrong about it as, according to the senior academic, AICTE regulations permit such projects with zero research component to be considered as M.Tech. projects. [BTW I tried to locate/download AICTE regulations for M.Tech. projects but did not get it.]

If, on the other hand, the student is found capable enough in the software engineering project work side by the supervisor, then the supervisor can, or perhaps should, advise the student to take up a more challenging research oriented project (which will also typically have some software engineering component). BTW I entirely agree with this view. This allows good and excellent students to get the best out of their M.Tech. project - research angle exposure as well as some software engineering project work.

But "actively encouraging" all M.Tech. students to take up research oriented project work in AICTE type of universities (unlike the elite IITs) can be ill-advised if the regulations allow for zero research software engineering projects. In my strong view, I consider it to be an academic ---- to brainwash M.Tech. students to do research projects (or scare them by veiled threats of poor marks for software engineering type projects) when the academic regulations allow for zero research software engineering projects.

The senior academic pointed out very well that if a department wants to have a Masters program which is only research oriented, there are AICTE approved programs like M.S. by Research which a department can take up. So no confusion from regulations point of view at all. BTW today I came across even an M.Tech. (Research) program.

Another point that came up in our discussions is about rural university/college issues. Typically such universities/colleges have significant number of rural students. Many of these rural students who enroll into an M.Tech.(CS) program may not really be strong enough technically to take up research work in their M.Tech.(CS) program. Expecting all such students to do research as part of the M.Tech. project work and then getting disappointed by the work they do seems to be a grave error of advice/judgement.

Yet another point is the need for senior academics to have research assistants for doing research work and publishing research papers. In places where senior academics do not have enough PhD scholars or M.S. by research students associated with them, the senior academics may view M.Tech.(CS) students as their research assistants! That, IMHO, is not an ethical practice.

---

Corr: A few clarifications:

1. copy-paste programmers are no good -- even today. It is just a matter of who identifies/interviews and makes that call decides who get in and who get out. In my 3 years, I only okayed one candidate. My managers learnt to ensure only high performer positions were interviewed by me too. Otherwise, they realized that it will simply be a time waste to try to sneak a candidate through me -- for good or for worse.

2. There is no excuse for admitting students to M.Tech.(CS) with poor programming skills -- even though you point out that this routinely happens. Perhaps if this problem is solved, the problem of having to deal with non-performing students during project phase will go away. But I guess the problem at hand is to treat the symptom rather than the root cause. Put another way, it is trading one problem for another, IMO.

Ravi: I believe the standard of software engineering project work in B.E./B.Tech.(CS/IT) streams in many AICTE colleges is not great at all. And it is some of these persons who will come into an M.Tech. Guys coming from other streams which do not have a software engineering project at all are also strong possibilities. They want to do an M.Tech.(CS) due to the job potential mainly, I guess, and so some colleges somewhere will provide the programs to meet the demand. As I see it, it is a simple supply and demand issue.

And so, in the foreseeable future in India, lots of M.Tech.(CS) entrants may have not so strong programming skills. The point is that even a free education system needs students - even if they don't meet a minimal standard. For commercial education systems, which are the vast majority of CS education systems in India, without sufficient students they may not be able to pay faculty salary!

These, IMHO, are the inescapable realities of non-elite CS/IT education in India.

Corr: 3. Yes, there is no excuse for forcibly thrusting research on people who can't cut it -- there is no arguing about that, IMO.

---

Corr later passed me two links referenced below.

The article, "How Different Is A B.S. In Computer Science From An M.S. When It Comes To Recruiting?", <http://www.forbes.com/sites/quora/2013/05/09/how-different-is-a-b-s-in-computer-science-from-an-m-s-when-it-comes-to-recruiting/> was very interesting to me. Some quotes/points that I found to be of particular interest:

\*) "In my experience, an MS degree has been one of the strongest indicators of poor technical interview performance." *[Ravi: That, from a US recruiter, is a very strong criticism of the MS post-graduates tech. skill levels in the US.]*

\*) The article mentions that some MS in CS students have an undergrad degree in some other field and so never do an algorithms or data structures class. It further states that graduate-level (i.e. MS) algorithms class would have easier grading when there are many students in it who have not done CS in their undergrad studies. *[Ravi: Of course, there would be some exceptions but when taken as a whole across the US, like in India, there must be many students doing MS with poor level/knowledge of programming/software engineering skills acquired in their non-CS graduate degree.]*

\*) It mentions that one of the attractions of an MS from a top CS school is the paper degree legitimacy value. It proposes another option of doing Udacity and Coursera classes, building some cool software and working with a recruiter. *[Ravi: Wow! That's some clear and off-beat advice. I don't know the US job market and the quality of Udacity/Coursera classes well enough to comment with certainty. On the face of it, the advice seems to be quite an interesting option.]*

"MSCS", <http://blog.regehr.org/archives/953>, was very interesting. Looks like MS coursework-only may be like Indian M.Tech. Don't know if the MS will have project work though. The research MS seems to be similar to MS by Research in Indian academia.

\*) This article references the previously mentioned article, "How Different Is A B.S. ..." and draws attention to the article stating that MS in CS has become a cash-cow. *[Ravi: I think the cash-cow point may be really hitting the nail on the head.]*

\*) "The best MS students and the best undergrads are extremely strong. However, the median-quality MS student is weaker than the median-quality undergrad. A lot of this is caused by the MS students who don't have a CS background: they simply are not ready for a serious upper-division CS class." *[Ravi: This time the comment is from a US Associate Professor and not a recruiter. So clearly, in USA, there seem to be significant number of MS in CS students without CS background who struggle to cope with Masters level course work.]*

\*) "At many institutions, the responsibility for not getting in over one's head is placed largely on the students. While this is in some ways admirable, it does lead to some amount of unnecessary suffering." *[Ravi: I read this as the poor student not getting a clear idea of what s/he is getting into by going in for MS without having a BS in CS background. And then s/he is left on her/his own to cope up with the Masters level classes! Unfair deal for the student, IMHO.]*

\*) *[Ravi: This article seems to be biased towards MS by research CS. The advice the author offers a student who intends to pursue an MS is research oriented. The author did not clarify that the student wanted to pursue MS by research - perhaps the student did not know much about differences between MS by research or coursework-only MS. Ideally the advice should have covered pros & cons of both MS options. IMHO, this is reflective of the strong research bias most CS academics have at the expense of technology skills.]*

---

Another correspondent who has taught CS in UK academia shared this comment with readers:

"In the UK they have what is called a 'conversion' MSc and an 'advanced' MSc. The first is for students from other disciplines who want to learn some CS; classes are very often at about the same level as final year degree CS courses. This kind of course is very popular with overseas students and hence with university administrations who can charge them higher fees.

There are fewer advanced MSc courses and they are usually intended as preparation for research."

# HRD Minister Javadekar says research not necessary for promotion for college teachers; university dept. faculty however need to do research

Associated blog post date: 30th July 2017, last updated on 27th Feb. 2020, link:

<https://eklavysai.blogspot.com/2017/07/hrd-miniser-javadekar-says-research-not.html> , short link:

<http://bit.ly/hrd-miniser>

Here is a report that appeared in The Hindu, a mainstream South Indian newspaper, today, College teachers may skip research, <http://www.thehindu.com/education/colleges/college-teachers-may-skip-research/article19386791.ece>, dated 29th July 2017.

Given below are the quotes of Union Human Resources Development minister (who is also in charge of higher education) Shri Prakash Javadekar, from the article:

"Making research compulsory for college teachers [has] harmed research. Thirteen thousand UGC magazines came up. Many colleges made their annual magazines into quarterlies and added them. I said there are so many journals here: do you have Champak too?" [Ravi: From <https://en.wikipedia.org/wiki/Champak>, "Champak (Hindi: चंपक) is a popular fortnightly magazine for children published by the Delhi Press Group since 1969 in India". end-Ravi]

"College and universities teachers are two different kinds of categories with different expectations. College teachers' primary responsibility should be to teach well. That accountability is required" .. "We will not make research compulsory for them. We will say, 'It is your choice'".

[Note: The article following this one in this book shares the audio link and a transcript link of the speech where above comments seem to have been made by Hon 'ble minister Javadekar.]

Ravi: I am so very happy to read these statements of HRD minister Javadekar. It has made my day!!!

Some years ago Indian UGC/AICTE governed academia, including Computer Science and Information Technology (CS & IT) academia, became diseased by the research publication craze as it had become mandatory for all academics for promotion. The impression I got was that faculty including junior faculty tried to publish research papers of any kind in all sorts of research publications. Typically, for significant number of faculty, the objective was not really to add to knowledge in a field but to get credit for publication of papers which would make them eligible for promotion! I further felt that the emphasis on teaching fell with negative impact in general on learning outcome of students graduating from Indian academia.

However, I did not find solid data to support my above views. So it was an unsubstantiated opinion.

But that has changed now. The union HRD minister who is the top man of higher education funding and regulation in India himself states that 13,000 UGC (research) magazines (publications) came up after research was made compulsory for college teacher promotion! Mind you, the number is the research magazines and not research paper publications which would be a big multiple of this number of research magazines.

The minister hints about poor quality of these research magazines by sarcastically asking whether a children's story magazine, Champak, was also added to the UGC research magazines list!

I am sure that the quality of most of the research articles in these 13,000 new UGC research magazines, in general, would have been typically from poor to junk. If so then not only were many Indian academic

teachers neglecting teaching but they were doing poor quality research and not genuine research, as they just wanted to publish some articles and get credit for promotion.

HRD minister adds that primary responsibility of college teachers is to teach well and that they should be held accountable for that. They surely can do research (but NOT at the cost of the primary responsibility of teaching well).

Fantastic!!! What a great message HRD minister has sent to college teachers in Indian academia. I think this will have a good impact on Indian CS & IT college teachers as well. I am so happy about this.

HRD minister treats university departments differently. The article states that university department faculty will continue to need research publication output to get promotions. Well, if universities ensure that its faculty do get enough time and support for good and genuine research, I support this research req. for promotion in universities. However, I know that in some private and deemed universities in India, junior faculty (teachers as against research scholars) are heavily loaded with teaching and other work of the department. These junior faculty are simply not provided much time for research. How then do these junior faculty manage? Either they have to neglect their teaching duties and focus on research or focus on doing a good job of teaching and do some research on the side whenever time permits.

I have personally seen how increased emphasis on research publication output and bagging high value research project grants, made teaching a less important activity in an Indian Computer Science deemed university department, especially among many young faculty. I was shocked to see teaching getting relegated to less important status and research publication output and bagging high value research projects becoming the most important activities of the department. The poor students did not really understand such changes as they blindly went through the system as decided by senior academics of the dept. Further, some M.Tech. (CS) students were brainwashed to make joint research publication with faculty a major first objective of their project work, when there are ABSOLUTELY NO SUCH MANDATORY REQUIREMENTS from AICTE norms for an M.Tech. (CS) project. I mean, some M.Tech. (CS) students became FREE research assistants for such research project work. I consider such tactics as grossly unethical and exploitative of students.

If students are fully informed of the real needs of an M.Tech. (CS) project as per AICTE norms, and without any coercion whatsoever, allowed to choose whether they want a research publication focused M.Tech. project or something else, then I think it is fine if somebody does an M.Tech. (CS) research oriented project.

## **HRD minister Javadekar tells truth about UGC AICTE academia problems; removes mandatory research for college faculty (but not university faculty); assures UGC AICTE reform**

Associated blog post date: 2nd Aug. 2017, last updated on: 5th Aug. 2017, link:

<https://eklavyasai.blogspot.com/2017/08/hrd-minister-javadekar-tells-truth.html> , short link: <http://bit.ly/hrd-minister>

I came across this very interesting speech by Hon'ble HRD minister Prakash Javadekar at a national conference on higher education organized by a teachers' union/orgn. on 29th July 2017. Javadekar talks in a

free and frank manner about the problems in UGC AICTE academia, how he proposes to solve them, and about other plans and initiatives taken by his HRD ministry to improve higher education in India.

[UGC, [https://en.wikipedia.org/wiki/University\\_Grants\\_Commission\\_\(India\)](https://en.wikipedia.org/wiki/University_Grants_Commission_(India)), <http://www.ugc.ac.in/> and AICTE, [https://en.wikipedia.org/wiki/All\\_India\\_Council\\_for\\_Technical\\_Education](https://en.wikipedia.org/wiki/All_India_Council_for_Technical_Education), <http://www.aicte-india.org/>, are the leading Indian govt. run academic regulatory (and academic funding, at least in the past) bodies of India.]

I found that his speech had a lot of content which is hard to find from other sources. Further, as the top man of Indian higher education, he gets to know a lot about the reality of Indian UGC AICTE academia. So I felt that I should make a social media post having transcript of appropriate parts of his mostly Hindi speech and then translate that to English. This post can then serve as a source of easily available data on the Internet for anybody interested in these matters. It took a lot of time and effort for me to do this work but I think it is a worthwhile additional small contribution of mine to improve teaching quality in Indian UGC and AICTE administered academia.

The transcript of most of the speech of Hon'ble HRD minister Javadekar given below covers the following (not in the same order):

- Attacks fake/pretence (pretense in USA English) research
- Demands good teaching and accountability from faculty; Student feedback will be one part of faculty accountability
- Announces plan to remove mandatory research (from API) for promotion of college faculty [Under graduate level teachers]
- Announces that research will continue to be mandatory for university faculty [Post graduate level teachers]
- Promises good rewards for good teachers
- UGC AICTE reform; Graded autonomy plan
- Good education system leads to country becoming rich and not the other way around
- Sustainable prosperity-hood for the country comes through innovations; Universities should become temples of innovation
- SWAYAM free online higher education with exams & certification; available via DTH TV (offline) too
- Details of how genuine research and innovation are encouraged by HRD ministry
- Permission given to students to form startup companies in hostel rooms; 600 such startups formed
- Over 500 engineering colleges closed in past four years due to students rejecting them; students see placement record and do peer review of colleges
- National Ranking Framework; 40,000 colleges today; Only good colleges will survive

*Ravi: I wholeheartedly support Hon'ble HRD minister Javadekar's proposed plan to remove mandatory research component from promotion criteria of college faculty [under graduate teachers] so that college faculty can focus on improved teaching and therefore better learning outcome for students. But I do have some concerns about problems of poor teaching in post graduate level in university departments not being addressed by his proposed change.*

The video (seems to be audio + still pic) of the speech which was used to make this transcript is: Speech of HRD Minister Prakash Javadekar at National Conference on Higher Education on 29-07-2017, <https://www.youtube.com/watch?v=8U2BjZ1oQ9o> (short link: <http://bit.ly/hrd-min-vid>), 30 min. 40 secs.

Note that the event is a conference on Higher Education organized by Akhil Bhartiya Rashtriya Shaikshik Mahasangh (ABRSM) which seems to be a teachers' union and/or organization.



From its about us webpage, <http://www.abrsm.in/about-us.htm> :

Akhil Bhartiya Rashtriya Shaikshik Mahasangh (ABRSM) is an organisation imbued with Indianness with an aim to propagate the ideology of Cultural Nationalism in the field of education and society. Along with safeguarding teachers' interests including their salary, allowances, service conditions and other facilities, the Mahasangh keeps in mind its national objectives, plans and executes the programmes of social concern and educational upgradation.

Accordingly, Akhil Bhartiya Rashtriya Shaikshik Mahasangh comprising the range from Pre-Primary to University level teachers throbbing with the sense of Nationality and Indian (philosophy) was founded in 1988. Now Mahasangh is a country wide organisation. Presently, 35 State level organisations and more than 50 University level organisations spread over 24 States are affiliated to it.

--- end short extract from About Us webpage of ABRSM ---

Ravi: It seems to me that this teachers' union/orgn. is directly/indirectly affiliated with the political party, Bharatiya Janata Party (BJP), and so the minister who is from the ruling BJP party seems to view this teachers' union/orgn. as a kindred union/orgn.

Given below is my attempt at English rendering of the transcript of most of the speech (but not all of it) by HRD minister Javadekar. Note that my English rendering may have a few errors but not many. My Hindi knowledge is reasonable but not excellent.

### **English rendering of most of speech of Hon'ble HRD Minister Shri Javadekar**

#### ***From 0:00***

This organization [*Ravi: referring to the teachers union and/or orgn. that organized the conference*] is one that works for the nation. And therefore I remember Dattopantji. Dattopant put the biggest idea (ideal) of [*Ravi: workers'/trade*] unions in this country. Dattopantji said, "We will work for the nation. And take full wages for the work." This one slogan itself creates a separate identity for us. To take the full wages for the work, we will stop work - this is what others say. Dattopantji taught us that we will work for the nation and we will take full wages for the work.

So this full wages have to be taken. And I am the education minister. I will say this that you too will get full wages - 7th recommendation [*Ravi: This seems to be a reference to the 7th pay commission recommendation which if implemented for govt. teachers would result in increase of salary & allowances*]. [Applause]

...

#### ***Around 6:15***

When I got involved in the discussions about API [*Ravi: Academic Performance Indicators which seems to play a key role in determining promotion for Indian academics in UGC/AICTE regulated educational institutions*] - then I always go in detail. But the opposition to accountability from some organizations - in their remarks it was seeming that some professors/senior faculty were opposing accountability. Well, 16 lectures have to be given [*Ravi: I think that refers to standard UGC norms for per week teaching load of faculty*]. So what account are we asking? Well, how many lectures were done, how many tutorials were done, how much time was spent in student activity. They do not want to give even this. And I find this difference in you that how should a new system of accountability be, (you said) we will give you an initial form (draft). What initial form (draft) you give, we will welcome it - this is what I would like to tell you. [Applause]. As this is what partnership is.

### ***From around 7:05***

One cannot say that we are not accountable. You can suggest that what accountability framework UGC has drafted is not correct in this respect and therefore the correct system should be like this.

And therefore today I would like to declare ... When I studied (the matter) it came to my attention that research being made compulsory for college teachers ... What happened after making research compulsory (for college teachers)? Research got finished! And UGC (research) magazines (count) become 13,000. So (they) came with that list to me. And many colleges took their annual magazine, made it a quarterly and added it to the research list. I said the list is so much, is there Champak [*Ravi: A children's stories magazine, <https://en.wikipedia.org/wiki/Champak>*] and related magazines (in it too)? [Laughter]

But I always hate hypocrisy/pretence and I do not want to live in hypocrisy/pretence. College faculty and university faculty are two different categories. They are two different jobs. There are two different expectations. In one you teach post graduates and in the other you teach (under) graduates. And therefore college faculty should have the primary responsibility of teaching well. That is the accountability. [Applause]

And the accountability will be that (you) teach well. And I am going to start a new procedure. I am [*Ravi: with emphasis on am*] going to start students' feedback mechanism. [Applause]

As accountability is not one sided. You said two sided. So I am doing it as three sided. So we will take student feedback too. But teach well - this is your primary responsibility. We will not put the burden of research being made compulsory on your forehead. We will say that it is your choice. You will have to surely do one student activity or one community activity. So your teaching + community is what will assure you promotion.

I will be bringing this change to API and we will declare this change in a short time. ... And from this platform of ABRS [*Ravi: Acronym of union/orgn. that has organized the conference*] I am declaring for the first time that we will make this change in API which will remove pretence/hypocrisy and college teachers/faculty will have teaching well and doing one student activity or community activity as their primary task(s).

And research will be compulsory for university teachers/faculty. And for those college teachers/faculty who have a proclivity/inclination for research, we will not finish their choice to do research. But that will be a choice. It will not be mandatory. Otherwise research becomes a joke. We should not allow that to happen. So this we will do.

...

### ***Around 12:19***

The ad-hoc teacher matter in Delhi is amazing. In all other universities ad-hoc teachers comprise ten to fifteen percent. In Delhi ad-hoc teachers particularly are the main force. Nine thousand ad-hoc teachers! I said we will stop this. We will make regular appointments. And this year I am behind that matter every month. In one year we will finish the whole process of regular appointment - I would like to say this too. [Applause].

In that those who are good ad-hoc ... There will be competition.

### ***Around 13:0***

Competition should never be opposed. See, whatever you may say about politics, every five years we have to compete to come (back to power). We do not come for free. [Laughter] In our place/profession [*Ravi: minister seems to refer to politics profession*] we do not have assured career promotion. Therefore you should not oppose competition on ... But you are experienced. As you have taught for eight to ten years, Indian CS & IT Academic Reform (Past) Activism Blog Book

naturally you will become permanent. You will get selected into permanent (category). Some may not become (permanent). This rule is also there.

Sometimes I think about this hire and fire (management approach) - I go to those countries. There nobody fires anybody actually. [*Ravi: I wonder whether he referred to Western world academia only as in regular companies, hire and fire does happen. Perhaps it happens in Western world academia as well but that was not noticed by the minister.*] Where there is hire and fire, there nobody fires anybody because every year there is a review. The fear of being fired leads to everybody being active. And everybody is on their toes. And everybody makes improvement in themselves. There is improvement in the country only because of these (attitudes/actions).

So sometimes to understand systems we need to open the doors of our minds. And therefore the government took the decision on it ...

As we are senior faculty/professors what study should we do? Do the study and let me know. What has happened in those countries which follow hire and fire (policies)? What were the good and bad results? Let somebody study and let us know.

To achieve quality in education, quality in higher education, the government has, as per Modiji's vision, started some remedial actions. I would like to only make a small mention about it.

Gargji said that in UGC reform - essentially he was saying that the rules/regulations of UGC have become very old. ... I told the officers. I am prepared to close down UGC. But we will need some administrative system. So what (will be) that administrative system? So what should be the reforms for UGC and AICTE? Have discussions on each rule/regulation and give me suggestions. Those suggestions are welcome. I would like to say that. [Applause] As we want to increase ... count.

And we have made one direction definite. That UGC and AICTE reform will be now graded autonomy. What is the meaning of graded autonomy? Those colleges which will go up in quality will get complete autonomy. Yesterday only, in the Lok Sabha [*Ravi: Indian lower house of Parliament*], unanimously [*Ravi: supported by all members/parties*], the IIM [*Ravi: Indian Institute of Management, the premier management institutes of India funded by the govt.*] bill was passed. This is a historic bill. This is a bill which gives complete autonomy to IIM. The government has no (role) in it. There is no need for (them) to go to the government everytime. This is the primary mantra of autonomy. This is how education evolves/grows.

The specialty of education in foreign countries, advanced countries is that countries have advanced only through progress in education (system). Like they say about roads that one day a team of representatives of India went and met America's president and told him sir, ... as you are wealthy your roads are good. Then the president said, no, no. It is because our roads are good that we have become rich/wealthy. Do understand this. So it is not that that country is rich and so its education (system) is good. The education (system) is good and so they have become rich. This is the truth.

And so if our education (system) is to be good then autonomy is the life force of UGC reform. And we will surely achieve UGC AICTE reform! And autonomy will be given to (good quality) institutions. Medium quality [*Ravi: literal words used were less quality*] institutions will get 50-50 (autonomy). And even lesser quality institutions that are at C level they will have regulations like (it is) today. But even in other regulations we need to do lot of improvement. Give those (to me/us). Taking each regulation - there are thousands of regulations - the UGC and AICTE system was like this that (the regulation) came in the dream in the night, the circular was put out in the morning and it got implemented the next day! I said this will not happen. First we will discuss. We will keep fifteen days for each ... for stakeholder consultation. ... We are opening the doors of online education. So to maintain the quality of online education we have made draft rules. We have kept those rules in front of you.

In future days any circulars about big changes in UGC AICTE will first have consultation (with) stakeholder(s) and then it will be given its final form. And if they are of (..) impact/effect then they will be Indian CS & IT Academic Reform (Past) Activism Blog Book

implemented only next year. In mid way (we) will not change the goalpost. This is also a definite rule of ours.

So we are proceeding with this reasoning and we need your co-operation.

### ***Around 18:12***

We are doing one thing. There are two or three initiatives (in this regard) by Modi government. Research and innovation. Because we lack in innovation. India lacks in innovation. And I have umpteen times said, that there are only three four ways to get country richer. And what are these. The one model which was adopted by Europe - Portuguese, Dutch, French, British - they conquered the world and looted the colonies - colonized the world and looted the colonies. And they became rich. But that's not a good way and now not available also. So that is one.

Second is the nature's bounty which the Middle East has. Wherever you dig you get oil .. That's a natural (resource) but that you can't demand. What you have, you have.

The third is India and China's way of last thirty years, of reverse engineering and cost-competitiveness on the basis of - essentially on the backdrop of - ah - low wages. We have achieved cost competitiveness and therefore we have dominated in exports, in world markets and .. China majorly and India to an extent, became rich with this kind of thing. But that's not a sustainable prosperity-hood.

The sustainable prosperity-hood comes through innovations. And therefore higher education centres must become - universities must become - temples of innovation. They must become centres of research (and innovation). If they are not - then if PhD students are applying for sweeper's job that means we have not taught anything. See, world over PhD is something addition to the existing knowledge.

We have made PhD so thin/shallow/superficial that any new interpretation through the existing knowledge is also PhD. It is not as if people are doing PhD for doing PhD [*Ravi: I think it means: 'It is not as if people are doing PhD out of desire to do genuine research which is what should be done for a proper PhD.'*] In our place/orgn./area we had a director. He had/owned very big educational institutions. He did a PhD. So we asked, how did you do PhD? I asked, in which subject/topic did you do PhD? So he asked (his) P.A., Hey, what subject/topic was that? [Laughter]

So this is not (full) quality education.

And therefore we started two three initiatives/experiments. We initiated/experimented with GIAN. [*Ravi: I think this is what the minister is referring to: "Global Initiative of Academic Networks", <http://mhrd.gov.in/gian>, <http://www.gian.iitkgp.ac.in/>.] In the previous year 600 senior faculty/professors from 58 countries came to India. They stayed for two to three weeks. They conducted a course in their subject. This year 800 senior faculty/professors will come. They will come from 70 countries. And (when) good senior faculty/professors come then our faculty/teachers also interact face-to-face with them. Our students also get knowledge [*Ravi: The Hindi word used is gian, also written as gyan, which means knowledge; so there is a pun here involving the acronym GIAN mentioned above.*] And we record all that, which we are also converting into SWAYAM courses. [*Ravi: The minister seems to be referring to the HRD ministry/govt's free online learning portal SWAYAM: <https://swayam.gov.in/>.] So they will come as SWAYAM courses too.**

A few days ago I had taken up a new initiative. Our national (chief) economic advisor, Prof. Arvind Subramanian - he has taught in John Hopkins and Harvard and best of the best - world's best universities. [*Ravi: The minister seems to be referring to Mr. Arvind Subramanian, Chief Economic Advisor to Indian govt. now, [https://en.wikipedia.org/wiki/Arvind\\_Subramanian](https://en.wikipedia.org/wiki/Arvind_Subramanian). Subramanian is a world famous economics expert. But he does not seem to be a regular academic professor. However, the wiki page mentions him teaching at Harvard University's Kennedy School of Government from 1999 to 2000 and at Johns Hopkins' School for Advanced International Studies from 2008 to 2010. end-Ravi].*

We are not utilizing the professor. I said you are professor. Give me one week. And he gave. And he conducted a course on contemporary .. issues of Indian economy and Indian economic survey. As we are giving our economic survey in a new way in the past three years. So he conducted a course on it and we appealed to senior faculty/professors across the country that those who want to attend the course to please register (for it). We called the first 200 senior faculty/professors who registered. Senior faculty/professors who teach economics came from Andaman [*Ravi: A remote island part of the country*] to all other parts of the country. And in front of them he gave thirty three lectures. Five lectures everyday. He conducted the course in 7 days. (That will now be available in SWAYAM).

### ***Around 22:30***

This year, there have been 1.5 lakh [*Ravi: one lakh is one hundred thousand*] registrations in SWAYAM. Our portal has become so popular. And there are over four hundred new courses available on it now. So we are quickly progressing on this new kind of platform of SWAYAM. This is (available) online as well as offline.

I am saying this so that you know who your competition is going to be from. I am also going to tell you that otherwise even if AC [*Ravi: Air Conditioner*] is fitted in classrooms, students will not sit (in those classrooms). Then they will study there (in SWAYAM). That is free. Because there will be two thousand courses there. It will be free. (There) will be lectures. (There) will be material. (There) will be tutorials. By the best of the best professors. (There) will be interactions. (There) will be discussion forums. Mid-term exams will be (there). Exams will be (there). Certification will be (there). These students will get choice based credit. And other learning (will be) made forever - If I [*Ravi: I think the minister is referring to himself*] want to learn something I can do so. I will get a certificate. So this system is new and it is online as well as offline on 32 DTH TV channels. As for this online is not required. For that [*Ravi: DTH TV*] only a set top box for free DISH has to be taken [*Ravi: I think the minister is referring to free Satellite TV/DISH TV services provided by Govt. of India for which a suitable set top box has to be procured/bought.*] For that, you spend 1300 Rupees once and you will get it permanently. (And people will learn this). We also can use this technology and these courses to - if you also teach the same (subject) then if I am the teacher I will say that students, listen to this lecture and come. We will run the class in a (flip) class model. We will do discussion here.

...

### ***Around 25:27***

There is a project "Ucchta Aavishkaar". In this Ucchta Aavishkaar project there is interaction between industry and academia. Hundred proposals - customized requests by industries have been offered and IITs teams of faculties and students together working on those hundred ideas. .. They will come out with the solutions. That is what higher education is all about.

Along with that we have created a program of IMPRINT. [*Ravi: I think the minister is referring to this: <http://imprint-india.org/>. IMPRINT acronym seems to come from the term: IMPacting Research INnovation and Technology.*] In IMPRINT, we called for research proposals from all professors, all professionals, all students and we received more than 1500 proposals in ten domain sectors and we finalized 200 proposals and we are funding 600 crore Rupees [*Ravi: 1 crore is 10 million*] for these 200 projects. Many things are on and now, they are into that research mode. That is what constructive engagement is all about.

I am not agreeable to [*Ravi: research done*] only for your API score. But those who on their own [*Ravi: initiative and interest*] do research, for them certainly they must get something [*Ravi: some reward*].

This is what takes the country forward. If this is not there then what will we do? We gave permission to begin startup in hostel rooms [*Ravi: I think the minister is referring to startup companies being formed in student hostels*]. And I am happy that the young students of India have the power of ideas and those students

who are rich/gifted [*Ravi: not money-rich but idea-rich*] with those ideas have started 600 startups from (their) hostel rooms. This is a powerful initiative. This higher education is going there.

Are we also with that or what are we doing?

Along with this we have also started incubation centers. And we now have three final initiatives - we created NRF - National Ranking Framework. Two years have passed. In the third year it will become perfect. The entire design.

But why NRF - good people - all people have started giving (it) - that our number in national ranking is this. So please come to our college. This is what should happen. That is the healthy competition. See, now without (good) quality, colleges will not run/survive - this I want to tell you today itself. And students will close it. Government will not close it. In the past five years, 500 engineering colleges in the country closed down. 500 - in four years [*Ravi: In the audio, somebody seems to have said something to the minister leading to him changing his words slightly*]. Over 500 colleges closed down. Who closed them? Students did! Students go to - they are now informed technically. They go to the website. They see the placement record. They go to the peer review. Do peer review. And ask those students, past students, alumni, how the college is? And if they don't get good report, they don't take admissions.

500 colleges closed! If some government had done that, by now we would have a (people's) movement against it. But students closed it. So what can (we) do? They [*Ravi: seems to refer to engineering colleges*] ask us for permission to close. We are giving that liberally - (we say) GO CLOSE DOWN (the college)! [Laughter]

As it is by this that quality (...). 40,000 colleges are there. Our idea/thinking is that maximum number of colleges should be good. And for this NRF is being used. NAAC ranking is being used. Now all of the facilities - grants, everything will be linked to the quality of the institute.

And then about many famous good colleges that are there, I have an experience about them too. That experience is this that these colleges are good because only good students come there. Therefore they are good. What has changed from your endeavour/efforts? This is what I want to see. And the system that I will create - you must have understood this much that I do understand something! [Laughter] So I will get to the entire details of this (in the system I create) [*Ravi: Literally the minister said he will prepare the entire horoscope of this, which figuratively means that he will get to the entire details in the new system he creates.*] [Applause].

What is the endeavour/effort of the college? What have you (done) to bring improvement, to develop the splendour and the potential of the student, what are the efforts/endeavour of the college? This (is what) we will see.

And this way we are creating a new construct. Lastly, we have started higher education finance agency. This is a new initiative. Why is India behind? Because .. Why do the best of the best 500 students of the country go abroad? They go because there (foreign countries) they have the best laboratories which are not there (available) here. We want to create best labs here. And for that in the coming three years, through the higher education finance agency, we will make an investment of twenty thousand crores [*Ravi: 1 crore is 10 million. So 20,000 crores becomes 200,000 million which is 200 billion (Rupees)*] in this field. So that with this the research infrastructure becomes strong.

Higher education finance agency has started (work). Next .. month the first loan will be disbursed (by it) for which there is no interest. It is (non-profit) in a way. But the habit that has formed about (asking for) grant for every thing, I want to (somewhat) get out of that too. Many improvements have to be made. People like you are there, are (there as) companions and so my enthusiasm increases further. You have many discussions over these two days and give a lot of suggestions [*Ravi: The event where the speech was made seems to be a two day national conference on higher education.*] Your suggestions are welcome.

Thank you very much. Victory to India [Ravi: I have translated Hind as India]! [Applause]

---- end English rendering of most of HRD minister Javadekar's speech ----

The blog post associated with this article has a transcript of most of HRD minister Javadekar's speech in the original Hindi with little parts in English that he spoke in.

*[I thank Hon'ble HRD Minister Javadekar and have presumed that he will not have any objections to me sharing the English translation of many parts of his speech in this book, whose ebook is freely downloadable/viewable by all, and whose paperback book is not-for-profit for me (author). This book does not have any financial profit motive whatsoever for the author.]*

## **Dr. Pushkar's articles on research req. for Indian college teachers producing fake research and reducing teaching quality; historical background of Indian academic research system**

Associated blog post date: 4th Aug. 2017, last updated on: 5th Aug. 2017, link:

<https://eklavayasai.blogspot.com/2017/08/3-interesting-articles-by-dr-pushkar-on.html> , short link: <http://bit.ly/3-interest>

I came across an interesting article that appreciated HRD minister Javadekar's recent announcement of near-future change in UGC/AICTE API (Academic Performance Indicators) to remove mandatory research requirement for college teachers (under-graduate teachers) - Kudos to Javadekar for Letting College Teachers Teach by Pushkar, <https://thewire.in/163562/javadekar-ugc-research-college-teachers/>, 1st Aug. 2017. The sub-title for the article is: "The research requirement in the API for college teachers was a travesty. All that it achieved was a proliferation of fake journals for college teachers to publish in."

About Dr. Pushkar who is Director of The International Centre Goa (India), from [archived link:

<https://web.archive.org/web/20170704190030/http://www.internationalcentregoa.com/web/about-icg/whos-who/> (short link: <http://bit.ly/2vlrqdp> ) for following broken link]

<http://www.internationalcentregoa.com/web/about-icg/whos-who/>: "Dr Pushkar (PhD Political Science, McGill University) previously taught at BITS Pilani-Goa, McGill University, Concordia University and Goa University." .. "He also writes regularly on India's higher education and his op-eds and essays have appeared in Asian Scientist, Indian Express, Hindustan Times, The Wire, Braingainmag.com and elsewhere."

Pushkar's above article refers to two earlier articles of his:

a) Let College Teachers Teach by Pushkar, <https://thewire.in/3617/let-college-teachers-teach/>, dated 10th June 2015

b) Strides Ahead in Sizing Up Academic Performance Skip Faculty Performance by Pushkar, <https://thewire.in/36677/strides-ahead-in-sizing-up-academic-performance-skip-faculty-performance/>, 18th May 2016

I feel that the three articles above give a good picture of UGC/AICTE Indian academia's struggle with mandatory research resulting in negative impact on teaching quality.

In the latest Aug. 2017 article, Pushkar himself summarizes his arguments for exempting (Indian UGC/AICTE regulated) college teachers (under-graduate faculty) from research, from his June 2015 and May 2016 articles mentioned above, as follows:

1. Large numbers of college teachers, even those with PhDs, lack basic training for research;
2. Most teachers are already overburdened with teaching, administrative and other responsibilities; and
3. A majority of college teachers work at institutions with woeful infrastructure and where the overall academic environment is inimical to substantive research.

Ravi: While I have not conducted a proper survey myself of Indian UGC/AICTE regulated under-graduate colleges, my views based on informal discussions on these matters over the past few years with Indian UGC/AICTE regulated college and university academics working mainly in South India, AGREE ENTIRELY with points 2 and 3 above.

As I am interested primarily in good teaching of Computer Science and Information Technology (CS & IT), especially the practice of software development, in UGC/AICTE administered Indian under-graduate AS WELL AS post-graduate academia, and have only some limited exposure to doing CS & IT academic research, I don't think it is appropriate for me to comment on point 1. Note that I have great respect and appreciation for GENUINE Computer Science and Information Technology academic (and industry) research and innovation, and as a software technologist I have applied/used the products/output of such research and innovation, to develop software and to teach software development in Indian industry and academia. But I am NOT INTERESTED in doing such research myself.

The Aug. 2017 article of Pushkar gives a very interesting extract from what it refers to as a widely-cited study (from the USA): The Relation Between Research Productivity and Teaching Effectiveness, by Herbert W. Marsh and John Hattie in "The Journal of Higher Education Vol. 73, No. 5 (September/October 2002)", <http://faculty.washington.edu/mpw/ITE05/research&%20teaching%20correl.pdf>. The extract is as follows:

It is important not to perpetuate the myth that there is a positive and reciprocal relation between teaching and research. There is no doubt that many would like such a positive relation to be true, and there is a strong conviction that research and teaching are closely linked...Furthermore, a near-zero correlation between teaching and research is consistent with the observation that some academics are gifted teachers and researchers, but that others are substantially better at one than the other, and some are weak as both teachers and researchers.

Ravi: I think the above effectively rebuts criticism that teachers who don't do research will be like drones who cannot apply knowledge well, and so will produce graduates and post-graduates who are like drones who cannot apply knowledge well. Mind you, like Pushkar says further in this Aug. 2017 article, teachers who are not trained to do research or are not inclined to do so, can keep up to date on research (in their field) and bring the new research input into their teaching if appropriate. I think this is like when I was an international software consultant in the Indian software export industry, I used to keep up to date on information about the latest technologies being offered by leading software vendors in the world, in my areas of software expertise, by reading leading industry/trade magazines on it regularly. And many times, I had to learn about some of these new software technologies so that I could provide consultancy advice about it, and sometimes even teach it to software developers. I did not have to do research into these new software technologies myself. Others did that and provided the new technologies. I consumed/used those new technologies to develop or help develop better software solutions for end users.

Pushkar writes, "It is not necessary, in terms of content, to be doing research in order to be or become a better teacher." I completely agree with this statement of Pushkar, based on my experience as an instructor/mentor/consultant in Indian software export industry for over a decade (different from my individual software developer and project lead experience), as well as a teacher of software lab. courses (mainly at M.Sc and at M.Tech post-graduate levels) and technical consultant for post-graduate (M.Tech./M.Sc.) software design & development projects in Indian academia for 9 years (with designations



of Honorary Faculty, Visiting Faculty and Honorary Staff). *[I also taught one software lab. course, once, to U.G. students (B.Sc. Mathematics).]*

Note that my academic teaching/tech. consultant experience of mainly post-graduate students was in a somewhat odd situation. The M.Sc. students I taught software lab. courses were Mathematics students (and Physics students a couple of times) and NOT Computer Science students. So while they were post-graduate students, they could not be viewed as post-graduate students from a Computer Science perspective. And most of these M.Sc. students later became students of the M.Tech. (Computer Science) lab. courses and projects that I taught/gave consultancy advice on. The important thing to note was that almost none of the M.Tech. (Comp.Sc.) students that I taught had a prior academic degree in Computer Science or Information Technology! Further, I believe that it is similar in many parts of Indian CS & IT academia. The M.Tech. (CS) or M.Tech. (IT) student usually has a NON Computer Science/Information Technology graduate (and sometimes post-graduate) degree! So he/she has to be taught some under-graduate level Computer Science/Information Technology subjects, including some under-graduate level software lab. courses, at M.Tech. (post-graduate) level!

The June 2015 article of Pushkar gives some interesting historical background of Indian (UGC/AICTE) academia. It says that in the "old system" there was a clear separation of teaching and research. Colleges and universities teaching at under-graduate level (offering graduate degrees to passed out students) are, Pushkar writes, "the backbone of India's HE system" (HE stands for Higher Education) with UG students comprising 80% of the total no. of Higher Education students. College faculty (and university faculty mainly teaching UG courses) focused on teaching. In the old system, a few college faculty did research and such people typically got promoted to/hired at PG institutions. But college faculty were not expected to do research with many not obtaining PhD degree or even seeking one.

In the old system, research was done by faculty in PG departments "across state universities, central universities and other central-government funded institutions". The PG department faculty had duties of teaching and supervising PG students (teaching-cum-research).

The article states that in 2010, the UGC introduced the API (Academic Performance Indicators) which made research mandatory for career advancement at central govt. funded institutions. This was also adopted by many state universities and colleges. The API was scrapped for a short time in 2013 but reintroduced (in 2013 itself).

This seems to have been an attempt to increase quality research output from (UGC/AICTE) higher education. Pushkar argues that it should have targeted only PG faculty and not UG faculty. By targeting all, it resulted in proliferation of fake research done only to secure API credit and reduction of teaching quality of UG faculty.

Now that the HRD minister is proposing to remove the mandatory research requirement for UG faculty/college teachers, Pushkar argues that it will allow UG faculty/college teachers to focus on teaching (Ravi: like it was before 2010).

*[I thank thewire.in, Dr. Pushkar and washington.edu, and have presumed that they will not have any objections to me sharing the above short extracts from, and many references to, their website/articles, in this book, whose ebook is freely downloadable/viewable by all, and whose paperback book is not-for-profit for me (author). This book does not have any financial profit motive whatsoever for the author.]*

# **IT Finishing Schools**

# IT Finishing Schools

Associated blog post date: 4th Oct. 2011, link: <http://eklavyasai.blogspot.com/2011/10/cs-it-academia-it-finishing-schools.html> , short link: <http://bit.ly/cs-it-acad>

A somewhat recent trend in Academia seems to be having IT Finishing Schools (ITFS) which help graduates of any discipline who could not land a job to pick up IT skills so that they land a job.

[http://www.nitt.edu/www/uploads/386\\_finschool.pdf](http://www.nitt.edu/www/uploads/386_finschool.pdf) specifies how Ministry of Human Resources & Development (MHRD), Government of India, directed NITs (National Institute of Technology) in Calicut, Durgapur, Jaipur, Kurushetra, Surathkal, Tiruchirappalli and Warangal and IIT-Roorkee to experiment with IT finishing schools. The document (dated 2007) also gives the syllabus covered in these "IT finishing school" courses.

And a finishing school set up by the Government of Kerala, states:

"Achieving quality standards in employability has fuelled the setting up of the Model Finishing School by the Government of Kerala. It is hoped that this venture will go a long way in remedying the perplexing problem faced by the youth of Kerala; of not being able to secure employment in desired numbers in spite of brilliant technical and analytical credentials"!!!

Here is the finishing school: <http://www.modelfinishingschool.org/>

I think these initiatives on part of the Central & State governments are wonderful, as it will be of great help to needy students who graduate with degrees & skill sets which are not in demand. I mean, it is all fine to be very knowledgeable about something and have "brilliant technical and analytical credentials" but if society does not have enough jobs/needs requiring that brilliant technical and analytical knowledge and skill set, how does such a person earn a living?

A great challenge in the Indian educational system is the availability of "higher education" opportunities for the masses which are relevant to society's needs, or in other words, relevant to job opportunities in the market place. So a student may do B.Sc. Physics only because he could not get a B.Tech. (IT) seat and not because he wanted to become a Physicist. Many times, he just wants to do his "graduation" as many white-collar jobs need at least a "graduate" degree. If the student had a choice he may have studied B.C.A. or B.Tech. (IT).

Now my comments should not be misunderstood as me looking down upon Physics or other sciences in comparison to IT. There are many brilliant students who do sciences, say B.Sc. Physics, out of an interest in Physics. But some, like me, studied Physics as I could not land a day-scholar engineering seat in Mumbai (in 1979), and could not really afford hostel-scholar engineering. It is not that I did not like Physics - it was just that I was given the impression then that the job prospects of engineering are better. And, given the financially strapped condition of my family, job was THE vital thing for me, and not knowledge for the pure love of knowledge.

[As an aside, even as a kid, the knowledge that I really loved most was scriptural knowledge but then there were, and still are, not many decently paid jobs/earning opportunities for guys who know Upanishads, Geeta or Bhagavatham! I mean the profession(s) related to scriptural knowledge expertise expect you to lead a life of Thyaga (sacrifice) and submission to God's Will, which at that age, frankly, I was not prepared to do.]

And even among those who study Physics (or other sciences or commerce or arts) as a second-option, like me, a love for Physics blossoms. IMHO, pure sciences like Physics are far, far more intellectually challenging to master than applied science streams like software/computer engineering. Specifically, IMHO, IT software design & development, while it does have its intellectual challenges & complexities, comes

nowhere close to the intellectual challenge of mastering areas of science like General Theory of Relativity ([http://en.wikipedia.org/wiki/General\\_relativity](http://en.wikipedia.org/wiki/General_relativity)) or (in Biology science field) understanding the genetic makeup of the human species ([http://en.wikipedia.org/wiki/Human\\_Genome\\_Project](http://en.wikipedia.org/wiki/Human_Genome_Project)).

Many of such science-first-choice or science-second-choice (or commerce or arts) students may become brilliant scientists (or experts in commerce & humanities) and contribute to extending the horizons of scientific (or commercial or artistic) knowledge.

But I think the inescapable truth is that many, many students doing science, commerce and arts degrees do it only to procure a decent job (like me). And IT is an area where many jobs are available. But IT degree seats are far, far fewer than the demand. So many students do something else as second-choice. And eventually many BUT NOT ALL science, commerce, arts and engineering graduates from various streams gravitate to the IT stream when they start looking for jobs. You just cannot escape from the market realities of Supply & Demand.

I feel, if AICTE/UGC policy makers seriously explore Internet based education possibilities, at least for the CS & IT field, they can really open up higher education opportunities to the masses. See the post on CS & IT: Internet Based Learning, <http://eklavyasai.blogspot.com/2011/10/cs-it-internet-based-learning.html> [short link: <http://bit.ly/2wt9PQU> ] for some thoughts on this.

Private finishing schools also seem to have joined the bandwagon. Here is a 2007 news report: [Archive link [https://web.archive.org/web/20090108034614/http://www.techtree.com/India/News/Mysore\\_Gets\\_IT-Finishing\\_School/551-79316-549.html](https://web.archive.org/web/20090108034614/http://www.techtree.com/India/News/Mysore_Gets_IT-Finishing_School/551-79316-549.html) (short link: <http://bit.ly/2SIDs9s> ) for following broken link] [http://www.techtree.com/India/News/Mysore\\_Gets\\_IT-Finishing\\_School/551-79316-549.html](http://www.techtree.com/India/News/Mysore_Gets_IT-Finishing_School/551-79316-549.html).

This 2007 news article gives some idea about the cost too: <http://www.rediff.com/getahead/2007/sep/24school.htm>

Another ITFS states:

"---- IT finishing school is a concept aimed at moulding industry-ready candidates. As per statistics, only 1 out of 4 engineers passing out from Indian Universities are employable. This is definitely a matter of concern and is a prominent factor that leads to unemployment among IT graduates. It may sound as a paradox that there are large volumes of unfilled vacancies in all leading MNCs. ---- IT finishing school is thus a creative solution for such bewildering factors. The new venture of ---- is an academic-industry association, where a student is provided training in all the needed hot skills and soft skills and then offered a real time industry experience."

It also says that it has "Offered service to more than 300 educational institutions in South India".

Here's its home page: [archive link: <https://web.archive.org/web/20110929234339/http://www.iprsr.org/finishingschool.asp> ( short link: <http://bit.ly/2v0Romj> ) for following broken link] <http://www.iprsr.org/finishingschool.asp>.

But does all this apply to CS & IT academia? Well, if CS & IT academia produce only "theory and research Generals" with poor programming skill set then employers from industry may prefer employing "software development soldiers" from IT finishing schools instead of CS & IT "theory and research Generals".

## More on IT Finishing Schools

Associated blog post date: 19th Oct. 2011, last updated on 4th Sept. 2014, link:

<http://eklavyasai.blogspot.com/2011/10/more-on-it-finishing-schools.html> , short link: <http://bit.ly/more-on-it>

[Archive link: <https://web.archive.org/web/20110211052850/http://www.careers360.com/news/3716-Job-hunting-The-truth-about-finishing-schools> (short link: <http://bit.ly/38Nbqzw> ) for following broken link] <http://www.careers360.com/news/3716-Job-hunting-The-truth-about-finishing-schools>: A very interesting article in Dec. 2009. Some notes from the article:

Dr K R V 'Raja' Subramanian, CEO of Radix Learning, a finishing school with a focus on IT and a former BITS Computer Science professor says 'We need outcome-based learning,' as that translates to 'job-ready' students

...

It is important for students to know their "unemployability quotient" and then (if that is high) focus on doing something about it.

--- end Notes ---

Here is a BBC article in March 2010, "Finishing school for Indian IT graduates":

<http://news.bbc.co.uk/2/hi/8547327.stm>.

Jul, 2009, Education Plus, Hindu article, "Applications invited for IT Finishing School": (20<sup>th</sup> Feb. 2020 update: seems to be either broken now or behind a paywall)

<http://www.hindu.com/edu/2009/07/13/stories/2009071350710800.htm>.

Here is the IT Finishing school mentioned in above article: [archive link:

[https://web.archive.org/web/20110807202244/http://teknoturf.biz/it\\_finishing\\_school.php](https://web.archive.org/web/20110807202244/http://teknoturf.biz/it_finishing_school.php) (short link: <http://bit.ly/3bSwZQR> ) for following broken link] [http://teknoturf.biz/it\\_finishing\\_school.php](http://teknoturf.biz/it_finishing_school.php).

[Archived link:

<https://web.archive.org/web/20110716015317/http://www.3edge.in/IT%20Finishing%20School.htm> (short link: <http://bit.ly/2HSyNMh> ) for broken link] <http://www.3edge.in/IT%20Finishing%20School.htm> Claims, "The concept of IT finishing school was pioneered in India by 3Edge in 2006."

<http://www.smartguruji.com/it-finishing-school.php> E-learning (online) course:

Mode : eLearning

Duration : 3 Months

Eligibility : Graduate/PG (Eg: BE/B.Tech/M.Tech, BCA, B.Sc., MCA, M.Sc. etc)

Fee : 4500 + Free TechnoFIRST Career Magazine (6 Months)

Syllabus:

- \* Programming Fundamentals
- \* User Interface Design
- \* C Programming

- \* HTML Basics
- \* LINUX OS Concepts
- \* Software Testing Basics
- \* Basics of UNIX
- \* OOPS Concepts
- \* Database Concepts (RDBMS, MS SQL Basics)

Placements: We have a 100% placement assistance program and you will also become a part of our extensive recruitment database after undergoing our 6 months industrial training.

....

Eklavya Sai (Ravi S. Iyer): Sounds like a pretty good deal. Wonder how good their placement record is? May not be great - sounds too good to be true.

-----

Well, the IT Finishing School concept seems to have become quite established in India. So looks like the division of labour is that AICTE/UGC governed CS & IT departments focus on the theory & research stuff and also impart some weak programming skill set, and the market-forces-created private "IT Finishing Schools" step in the "gap" to provide "job oriented skills" to those CS & IT graduates who could not land a job. Of course, some of these finishing schools are open to all graduates not just CS & IT graduates - but clearly many CS & IT graduates are not able to land a job without the help of such "IT Finishing Schools" or private training institutes.

**Academic PhD Club**  
**&**  
**Bureaucratic POWER structure**

# CS & IT Academia: The PhD Glass Ceiling

Associated blog post date: 8th Oct. 2011, link: <http://eklavyasai.blogspot.com/2011/10/cs-it-academia-phd-glass-ceiling.html> , short link: <http://bit.ly/phd-glass>

When I moved to Academia, after 18 odd years of international software design & development experience to offer Free Service (Seva) in the area of teaching Lab. courses in academia, I was in for a lot of 'learning' about CS/IT academic system.

One of the first things I learned is that Academia is essentially a PhD club. I mean, if you do not have a PhD you are automatically considered to be a less intelligent form of life and there is a glass ceiling, or rather glass partition much lower than the ceiling, which you cannot cross if you do not have a PhD.

As I had come with the intention of serving society as a form of God, I felt I should be humble and take such attitudes in my stride. As I was an M.Sc. (Physics) drop-out, i.e. only a B.Sc., it was as if I should be thankful to have been offered a chance to teach. My 18 odd years of international software experience did not matter all that much except that it permitted me to be a teacher of 'Lab' courses only and also to help M.Tech. project students with their programming problems. I later learned that this attitude stems from AICTE/UGC norms for appointing CS/IT university teachers. So I understood this to be an academic administrative issue and did not hold anything against CS/IT academia. I was happy to share my knowledge, for free, with students and that was an excellent arrangement both for me, and I guess or hope, the students :).

I think the PhD glass ceiling is a systemic issue - I mean, the AICTE/UGC governed system expects you to prove your intellectual calibre by doing a PhD. Once you have crossed that bar - then you are into the academic club for life. If you hang around in academia without doing a PhD, the AICTE/UGC governed system presumes that you are not good enough to do higher things in academia - you can hang around as a junior teacher (Asst. Prof. provided you clear NET/SET/SLET or just T.A. otherwise) but really don't deserve to go up the ladder. How good a teacher you are does not really matter!!! For the system, what matters is whether you have a PhD. That's it.

And this then naturally shapes the mind-set of academicians. Students who go through the system - Bachelor's degree, Master's degree(s), PhD - grow in theory & research knowledge but without really having any real practical experience of CS/IT! And, once they get the PhD they become Dr-name-members of the academic club! That gives immense recognition in the academic system.

To acquire a CS/IT PhD you have no need to be exposed to CS/IT industry world. So most of the academic CS/IT PhD club guys have no idea about academic qualifications of Steve Jobs or Bill Gates or Mark Zuckerberg and how it is their on-the-job study and work that allowed them to create great software products that have had such a significant impact on the world. Many would not even have heard of Mark Zuckerberg (though they would surely know about Facebook).

And most of the CS PhD guys generally are not that great in programming & design! Surprised and shocked, are you? Well, that's the reality. For research you need to break your head studying research papers and coming up with innovative approaches to your research problem. Of course you need an algorithm/programs to test your innovative approach to your research problem and get publishable results. But what matters is the result. How you get the results is not so important. You could use MATLAB scripts. You could write badly designed and unreadable C code but which somehow works and gets some results. That is perfectly fine. The paper that the PhD scholar sends to a journal does not include source code. Nobody who matters, cares about your source code - that is just menial-code - they care about the results that your code produced. That's it.



So the system has ABSOLUTELY NO INCENTIVE for a CS PhD guy to do design & programming part of his PhD work well. And then this is the guy who after getting his PhD, according to AICTE/UGC norms, is fit to be a teacher of design & programming as well as any other CS/IT subject under the sun to CS & IT students!

Okay, that's fine - that is the AICTE/UGC system. What can you do? They are the bosses who lay down the rules and academia has to follow them. Period.

But in the narrow world of CS/IT academia, some of these PhD qualified guys truly believe they know more than any of the CS/IT industry guys. They look down at industry as doing 'menial labour/coding' jobs. They use Google for their research, gmail for their mail, Microsoft Word for their documents and iPod/iPhone products as well but still look down at industry!!! And the poor industry experienced guy who does not have a PhD who tries to argue with them on teaching CS/IT stuff is treated with either disdain or with "he does not know academic issues" attitude. The knowledge level does not matter, what matters is whether you have a PhD - AICTE/UGC norms rule!

I would like to repeat that this PhD club mind-set problem among CS/IT academicians is not universal. But some, if not most, CS/IT academicians certainly have it in good measure.

I would like to clarify that what I have written above does not mean that I do not respect researchers! I certainly respect their endeavour to push the boundaries of knowledge in their chosen fields. Their life their choice! However CS/IT research does not appeal to me. What appeals to me is applying established technology to provide working software solutions that help society - a technologist's view perhaps as against a researcher's view. My life my choice! Both technologists and researchers are needed in society.

I respect researchers and I think it would be fair if researchers & academicians respect me and technologists like me, and don't look down at us because we don't have or are not bothered about acquiring a PhD! But if they don't respect me and other technologists - that's fine - it is a free world - they have the freedom to be arrogant - and I have the freedom to ignore them and steer clear from them. If, on the other hand, they respect technologists, I and other technologists, I am quite sure, would be happy to work with them so that we have some Research & Development (R&D) happening instead of only Research publications being produced from their work.

Unfortunately AICTE/UGC CS/IT policy makers are extremely pro-research & pro-PhD but do not provide enough encouragement for the technologist's view. That perhaps is the reason why we don't have any significant software to tackle country's problems being produced from CS/IT academia in the country. I mean CS/IT academia produces research publications - that really is it - and maybe there is some talk about some software but nothing really materializes and reaches the public.

## **CS & IT Academia: A Bureaucratic POWER structure**

Associated blog post date: 8th Oct. 2011, link: <http://eklavyasai.blogspot.com/2011/10/cs-it-academia-bureaucratic-power.html> , short link: <http://bit.ly/bureaucratic>

The POWER structure in CS & IT Academia is very different from the POWER structure in international software consultancy companies.

Academia has an essentially bureaucratic power structure. I think that must be flowing down from the Union Ministry of Human Resources & Development as they are the ultimate power centre for academia. They

give out the Grant money that flows down to universities and they control government recognition via AICTE/UGC (& status/reputation via NAAC).

For an academician, the HOD is the KEY POWER CENTRE. Massive amount of power is concentrated in him/her. Of course, academia gives a lot of room for academicians to explore their varied interests. And most HODs having come from the academic setup do give that freedom to members working under them. But if, for some reason, one gets into a direct conflict with the HOD, life can become very tough. You have to either get an amicable resolution to your conflict with the HOD OR you have to quit and join CS department of another college/university! I mean, if one were a PhD in CS, one cannot shift from a CS department to Physics or Management, for a paid academician post (free service is different as there may be some common/related courses). And to join CS department of another university, there has to be a vacancy there! Universities & colleges are far fewer than software companies! And the university/college with a vacancy could be in a different city/region and so, if you were married with children, the whole family will have to consider moving to another place!

So most academicians are very reluctant to CROSS PATHS with the HOD. If the HOD blocks something, it stays blocked. Efforts in the department which do not have the blessing of the HOD will typically wither away and die.

There is no HR manager to go to for listing out your litany of woes, who, if you are a good performer, will do his utmost to resolve the problem. That is a key function of the HR manager in industry - the company does not want to lose good people to the competition. In academia, you can find some Sympathetic Listener (Agony Aunt type) and unburden yourself - that is the nice thing about academia - generally the teachers are nice people, so they will listen to you and give you some peaceful and gentle advice - and wish you well. But they cannot change the system - you should not have a serious conflict with the HOD - if you do, you will suffer. Fact of Academic Life! Period.

In tremendous contrast, the software industry is really cool on that front. If you are good and you get fed up with one manager, you look around and shift to another group in the same company (using HR manager's services at times) or get another job. Jobs are no problem for good guys - yes, the pay may differ here and there and working environment may be different. But at least you can get another job and earn enough to run your household. And in the same city, or same suburb or even same building - no kidding - I was working with a company on the 3rd floor of a building in SEEPZ, Mumbai, the original "IT park" of India - moved out of SEEPZ to try freelance training instructor work - came back after a few months to SEEPZ by joining a company on the 2nd floor of the same building as a 'consultant'! (For more on SEEPZ see footnotes).

So if you are an industry guy and if you have a rosy, rosy picture of academia and have visions of, at some point in your life, sharing your knowledge with young students in academia and, perhaps, becoming a respected & revered CS/IT Professor, be aware that there are many thorns too. I am not saying that CS/IT Academia is all bad - no, not at all. One of the great joys of being in Academia is the joy of imparting knowledge to eager, and many times, grateful young students. That is a very satisfying experience. Industry may not be able to provide much opportunities for such kind of joy. The fellow teachers are also usually a very nice & friendly lot. There is a lot of respect from society too. And one works in a far more relaxed environment as compared to the typical industry environment. But academia certainly has its bureaucratic power structure and you have to fit into that very-different-from-software-industry power structure.

Notes:

Here are a couple of links on SEEPZ, Mumbai: <http://en.wikipedia.org/wiki/SEEPZ>, <http://www.seepz.gov.in/> - For most of my software industry career, from 1984 to 2002, I was based in SEEPZ - 4 different companies but all in SEEPZ. During the initial years of the software export boom, SEEPZ, Mumbai was where a lot of the ACTION was happening.

A wiki page states:

India's IT Services industry was born in Mumbai in 1967 with the establishment of Tata Group in partnership with Burroughs. The first software export zone SEEPZ was set up here way back in 1973, the old avatar of the modern day IT park. More than 80 percent of the country's software exports happened out of SEEPZ, Mumbai in 80s. Source:

[http://en.wikipedia.org/wiki/Information\\_technology\\_in\\_India](http://en.wikipedia.org/wiki/Information_technology_in_India).

*Wiki References: 6. "Special Economic Zones: Profits At Any Cost". Doccentre.net. Archived from the original on 7 September 2010. Retrieved 22 July 2010.*

*7. "Top 50 Emerging Global Outsourcing Cities" (PDF). [www.itida.gov.in](http://www.itida.gov.in). Archived from the original (PDF) on 21 September 2018. Retrieved 22 July 2010.*

-----

Given below is a comment from blog post associated with this article.

Ravi S. Iyer wrote on December 30, 2012 at 2:14 PM:

Here's an interesting article about vast difference between Western universities & Indian ones, Comparing Harvard apples with JNU oranges, <http://www.thehindu.com/opinion/op-ed/comparing-harvard-apples-with-jnu-oranges/article4242153.ece>.

A point which fits in very well with my experience of Indian CS academia is the following: "In fact, anything new is looked at sceptically, and often succumbs to the tyranny of age. Age-related hierarchy is perhaps the worst in the Indian university system and the least-debated sacred cow."

## **The very strange case of 20th century era Department(s) of Mathematics and Computer Science in Indian academia in today's early 21st century world**

Associated blog post date: 21st Apr. 2014, link: <http://eklavyasai.blogspot.com/2014/04/the-very-strange-case-of-20th-century.html> , short link: <http://bit.ly/the-very>

I think the importance of computers in today's early 21st century world is unquestioned by any realistic individuals anywhere in the world. However it is very strange that some Indian academic Computer Science departments (at least one that I am aware of) continue to live in a 20th century era. Specifically:

- a) They combine Mathematics and Computer Science as a single department - Department of Mathematics and Computer Science. Surely, in the early 21st century computer and Internet age, Indian universities should have a separate Computer Science department instead of combining it with Mathematics! I know of one such combined CS and Mathematics department case for sure. I do not know whether that is an exception with no other Indian university today having a combined Mathematics and Computer Science department.
- b) They do not offer undergraduate and immediate post-graduate degrees in Computer Science or Computer Science & Engineering. Instead students who are interested to acquire a computer science qualification from that university have to, in the normal case, first do B.Sc. (Mathematics), a 3 year programme (done after

12th grade), followed by M.Sc. (Mathematics), a 2 year programme, and then take up M.Tech. (Computer Science), a 2 year programme (a total of 7 years after 12th grade to get the CS qualification)! [The exception case is doing undergraduate Computer Science degree elsewhere and then trying to join the M.Tech. (CS) programme in this university after clearing their entrance exam & interview.] In today's age where students are ambitious, most students who want (or are constrained in some way like financially) to do Computer Science in a science, commerce and arts (UGC) university (as against an engineering or technology university) will pursue B.Sc. (Computer Science), a 3 year programme, I presume like other science degree programmes, immediately after 12th grade, and optionally follow it up with M.Sc. (Computer Science), a 2 year programme. [In engineering/technology (AICTE) universities such students will pursue B.Tech. (Computer Science & Engineering) or B.E. (Computer Science & Engineering), a 4 year programme (done after 12th grade), followed optionally by M.Tech. (Computer Science & Engineering) or M.E. (Computer Science & Engineering), a 2 year programme.]

Please note that I am discussing only Computer Science and Computer Science & Engineering branches in this post and not Information Technology or Computer Applications branches.

# **Indian Engineering Colleges: Graduate Employability Problems**

# Private study claims: Less than 9% of Indian engineering students have required programming and algorithm skills for IT product company jobs

Associated blog post date: 16th July 2014, link: <http://eklavyasai.blogspot.com/2014/07/private-study-claims-less-than-9-of.html> , short link: <http://bit.ly/private-s>

Here is a now-normal, but still depressing, article in The Hindu, dated July 16th 2014, Less than 20 per cent engineers are employable for software jobs: survey, <http://www.thehindu.com/todays-paper/tp-features/tp-opportunities/less-than-20-per-cent-engineers-are-employable-for-software-jobs-survey/article6214929.ece>

It is based on the Aspiring Minds, National Employability Report for Engineers - 2014, which can be downloaded (after providing contact info.) from here: <https://www.aspiringminds.com/research-reports/national-employability-report-for-engineers-2014/> (short link: <http://bit.ly/2T3ylzS> ).

Two small extracts from this (NERE-2014) report:

Page 9:

"The report is based on a sample of more than 1,20,000 engineering students from 520+ engineering colleges across multiple Indian states. All these candidates graduated in 2013."

Page 12:

"The employability of engineers in IT product companies is exceptionally low, to the order of 3.21%. This is because jobs in IT product companies require a strong understanding of computer programming and algorithms. The study found that the candidates strongly lacked the required skills: around 91.82% of graduating engineers do not have the required programming and algorithm skills required for IT product companies, whereas 76.23% show lack of soft-skills and cognitive skills."

--- end small extracts ---

Ravi: This report which says it is based on an extensive sample, tests programming and algorithm skills, states that less than 9 % of engineering-student-candidates (presumably final year students) got the grades/marks considered suitable for IT product company jobs! I think AICTE (India's top technical education regulator) and NPTEL/IIT-Madras (leading educational institutions) should have a similar computer programming and algorithm skills exam as a stand-alone exam which any final year engineering/science student/graduate can take. Then we will have two sources for such skill level figures with one coming from the top agencies/institutions associated with technical education in the country.

Now, pundits may dismiss this dismal less than 9% qualifying figure as being private survey results and question its methodology. If AICTE and NPTEL/IIT-Madras is involved in this testing then they cannot dismiss it so easily.

Further, students and parents should have access to the results of such exams for each educational institution. That will give them some idea of the teaching quality of computer programming and algorithms in these educational institutions.

TRANSPARENCY - that is what is desperately needed in Indian CS & IT academia. However, until Indian CS & IT departments of academic institutions are forced to share such information with the public, most will refuse to do so. So AICTE must take the lead and force such transparency on these departments/institutions.

-----  
A correspondent responded (slightly edited; NASSCOM is the premier Indian software industry organization):

Despite what the Hindu article says, according to NASSCOM around 300,000 of the million or so fresh engineering graduates do get jobs in the software industry. Of course, these jobs can range from mainstream developers to people working in the BPO industry.

NASSCOM 's figures are credible because if you add the number of new recruits claimed by the big 3-4 software companies you easily get to 100,000+. The other 200,000 could well find slots somewhere else in the industry.

However, standards are indeed dropping and the big companies have to run 3-6 month training programmes (read this as remedial training) before the new recruits can be moved to project teams. Even so, these companies have between 15-30% of their staff 'on the bench', or unbilled, which is an enormous cost.

--- end correspondent response ---

My response referred to Page 11 of the report which states that employability of “Software Engineer – IT Product” Role was 3.21% but employability of “Software Engineer – IT Services” Role was 18.43%. Employability of “Associate – ITeS Operations (Hardware and Networking )” Role was 35.37%.

So there is a differentiation between IT Product and IT Services. 18.43 % of 1 million (1,000,000) would be 184,300 jobs. And 35.37 % would be 353,700 jobs

Considering that the NASSCOM figures included the BPO jobs, we have to apply the ITeS Operations figure of 35.37 % (353,700 of 1,000,000) - so the figures do not diverge very much.

BTW the report claimed that 600,000 engineers graduate every year. But for the percentage figures, given that the sample size was over 100,000, it does not matter whether the total size is 600,000 or 1,000,000.

...

I guess the "employable" criteria of this assessment would be such that a person who qualifies does not need remedial training.

I guess the Indian IT services industry realized long ago that most CS & IT graduates, let alone graduates from non-CS&IT streams, would need remedial training given the poor standards of teaching of programming & algorithms in Indian CS & IT academia, and so took that burden on itself.

However, many IT product company types are now an important and excellent-salary employer of CS & IT graduates, especially in Bangalore & Chennai (going by what my past students tell me). They do not have any scope (or interest perhaps) for organizing remedial training and so graduates who are taught programming & algorithms better in colleges itself, get a break in these companies. And those graduates & post-graduates (like M.Tech. (CS)) who are poor in programming & algorithms fail to make it, even if they have published some conference research paper as part of their M.Tech. work. I mean, the bottom line for many of these product company types is programming & algorithms; research is extra.

[In contrast there are companies who seem to be looking for research assistant types and are happy to employ M.Tech. (CS) chaps as research assistants with some fancy designation. But, without a Ph.D., most of these chaps will surely hit a glass ceiling and get stuck doing lower-quality research work rather than great research work in these firms, IMHO. I mean, if you want to pursue research as a career a Ph.D. OR enough research publications to be counted as equivalent to Ph.D., is an absolute must.]

## **GATE and UGC NET CS & IT exams can be employability measures if they include practical knowledge assessment**

Associated blog post date: 29th June 2014, link: <http://eklavyasai.blogspot.com/2014/06/gate-and-ugc-net-cs-it-exams-can-be.html> , short link: <http://bit.ly/gate-and>

I made the following comment (slightly edited) to a recent post, How well does a college teach its students?, <http://www.obvioustruths.blogspot.in/2014/06/how-well-does-college-teach-its-students.html> (short link: <http://bit.ly/2V942KE> ), by Srinivasan Ramani, <http://internethalloffame.org/inductees/srinivasan-ramani> :

Interesting views. More transparency in Indian academic system in general will do wonders in giving students and parents some reasonable picture of the teaching and research quality of Indian academic institutions. As of now, most Indian academic institution websites do not carry significant information on course material related to courses taught by its faculty, though many carry the academic's research publication list. I think if the course material for courses taught by academics is put up on their website, students, parents and even employers will be able to get some idea of what is really taught in those courses. In great contrast to Indian academia, many US academic websites have detailed course material - Indian academia should follow their lead in this regard, IMHO.

Regarding standard exams that assess how well a college teaches its students, for engineering in India, we have the GATE exam, which is quite widely accepted by academia (for further education like M.Tech.) as well as government employers as the key measurement criteria for knowledge level of engineering graduates. Perhaps it would be a great idea to have its results available in the public domain but there may be moral and legal issues related to protection of privacy rights of students who got poor scores. The wiki page states, "The score cards are issued to only the qualified candidates."

As somebody who is interested in improving the practice of software development in Indian CS & IT academia, one issue I have with examinations like GATE, [http://en.wikipedia.org/wiki/Graduate\\_Aptitude\\_Test\\_in\\_Engineering](http://en.wikipedia.org/wiki/Graduate_Aptitude_Test_in_Engineering), is that they, I believe, focus on the theory part of the knowledge of the candidate, as that may be easier to assess. Even the National Eligibility Test for Lecturers (Asst. Professors) for UGC/AICTE regulated colleges, [http://en.wikipedia.org/wiki/National\\_Eligibility\\_Test](http://en.wikipedia.org/wiki/National_Eligibility_Test), for the CS & IT field does not seem to have a practical component for the test! What is badly needed by youth in colleges nowadays is employability, and employability needs the right blend of theory and practical knowledge. Once our national examinations including GATE and UGC NET improve their assessment techniques to properly assess practical knowledge then they may become an important measure of the employability of engineering college graduates.

## **The Hindu's higher education student guidebook - thenxt.step 2013 - CS & IT picture**

Associated blog post date: 20th April 2014, link: <http://eklavyasai.blogspot.com/2014/04/the-hindus-higher-education-student.html> , short link: <http://bit.ly/the-hindus>

A few days ago I saw an ad. in The Hindu newspaper about a sort-of student guidebook on higher education from The Hindu, thenxt.step, and decided to check with my town (Puttaparthi) Hindu vendor whether he had it. He did and gave it to me (for Rs. 250/-) but it turned out to be last year's issue! I decided to return it but



later changed the decision as I felt I might learn something related to higher education in Computer Science and Information Technology (CS & IT) from the 2013 issue itself (which, I believe, happens to be the first issue of this sort-of student guidebook on higher education from The Hindu). My changed decision turned out to be the correct one.

I have learned a lot about Tamil Nadu higher education sector in the areas of Computer Science and Information Technology from thenxt.step 2013, (broken link) <http://hindu.com/books/nxtstep/nxt13.htm>.

[*Tamil Nadu is a southern state of India with Chennai (Madras) as its capital,*

[http://en.wikipedia.org/wiki/Tamil\\_Nadu](http://en.wikipedia.org/wiki/Tamil_Nadu).] I have tried to put down some points from that learning below:

Page 3 has a full page advertisement of Vellore Institute of Technology, <http://www.vit.ac.in/>, which, I understand, is one of the leading private deemed universities of South India with its Computer Science and Engineering programmes being accredited by ABET Inc., USA. This ad. lists the following programmes in CS & IT at its Vellore campus:

#### **School of Computer Sciences and Engineering (SCSE)**

- B.Tech. Computer Science and Engineering
- B.Sc. Computer Science
  
- M.Tech. Computer Science and Engineering
- M.Sc. Computer Science

#### **School of Information Technology and Engineering (SITE)**

- B.Tech. Information Technology
- B.C.A. (Bachelor of Computer Applications)
- B.Sc. (Multimedia and Animation)
  
- M.Tech. Information Technology - Networking
- M.Tech. Software Technology
- M.Tech. Software development and Management (for employees of Cognizant Technology Solutions)
- M.C.A. (Master of Computer Applications)
- M.S. Software Engineering (5 year Integrated Programme)
- M.S. Information Technology (for employees of WIPRO Technologies)

---- end advertisement info ----

Well, that's quite an array of CS & IT undergraduate and postgraduate programmes. The programmes meant for a particular software company are quite noteworthy even if there may be strong arguments against having such programmes in a regular academic institution (as against an industry dedicated training institute which, however, may not be empowered to award bachelor or master degrees - at least during my programming learning days in the mid-80s they were not allowed to do so; they could simply provide some diploma certificates which were probably not recognized by the government then).

...

In an article by Dr. Hanifa Ghosh, principal CTTE College for Women, Chennai on Page 10, Dr. Ghosh warns parents against forcing children, who are not interested and/or do not have the aptitude for Indian CS & IT Academic Reform (Past) Activism Blog Book

engineering, to take up engineering in colleges with poor or uncertain reputation due to availability of seats there. She writes that they may end up doing poorly paying jobs unrelated to their engineering degree. Opportunities provided by Arts and Science colleges may be more appropriate for such students, she writes. ...

Jayaprakash Gandhi, a career consultant and analyst, writes in his article on Page 14 about it being better to choose a primary field (like mechanical engineering) instead of a secondary field (like automobile engineering) as the latter narrows down future career options at the initial stages itself. Further, he says, GATE exam (for entrance to IITs) are for primary fields and not secondary fields. Higher studies in India and abroad also may be easier to pursue for those who have studied primary fields.

*[IITs are the elite technical education institutions of India with substantial, if not total, government funding, [http://en.wikipedia.org/wiki/Indian\\_Institutes\\_of\\_Technology](http://en.wikipedia.org/wiki/Indian_Institutes_of_Technology). "The Graduate Aptitude Test in Engineering (GATE) is an all-India examination that primarily tests the comprehensive understanding of various undergraduate subjects in engineering and science.", [http://en.wikipedia.org/wiki/Graduate\\_Aptitude\\_Test\\_in\\_Engineering](http://en.wikipedia.org/wiki/Graduate_Aptitude_Test_in_Engineering).]*

Ravi: Perhaps the same primary field and secondary field argument applies to CS & IT fields. The Computer Science & Engineering field is a primary field with all the benefits of primary field mentioned above. In my browsing I have not come across definitive explanation of the Information Technology field in Indian academia and its difference from the Computer Science & Engineering field. Some years ago I had asked this question of a senior Indian academic who then was the head of the Information Technology department of an engineering college in South India. He told me that the subjects covered were almost the same (perhaps he actually said they were the same) as the Computer Science and Engineering (CSE) department of his college (they seemed to have that department too). The actual reason for having two programmes was related to some regulations which limited the no. of seats in one programme - having two separate programmes in CSE and IT was a workaround!

The B.C.A. and M.C.A. programmes being computer application programmes would probably be viewed as a secondary field with all the disadvantages mentioned above for secondary fields. However, the advantage of these programmes may be that they are less rigorous since they aim at imparting applications level skills rather than fundamental computer science skills, and so some students who cannot handle CSE programmes may be able to handle the computer applications programmes. And, very importantly, there may be industry demand for such computer applications skills which ensures jobs for BCA and MCA passed out students.

...

Pages 34, 36 & 37 carry a superb article titled, "Admissions demystified", by Prof. V. Rhymend Uthariaraj, secretary TNEA and professor and director of Ramanujan Computing Centre, Anna University, Chennai, which explains the single-window counselling (admission) process for the government seats of engineering colleges in Tamil Nadu. I strongly recommend that this article be read by anybody interested in understanding the engineering admission counselling process in India (other progressive Indian states would be following a somewhat similar procedure, I guess). The article states that Tamil Nadu's engineering college admission process is used by more than 1.5 lakh (one hundred and fifty thousand) engineering student-aspirants to get enrolled in Tamil Nadu engineering colleges.

The concluding part of the article is interesting and idealistic sort-of advice in general but not necessarily perfect and verified-to-be-fully-truthful advice especially to youth desperately seeking higher education that will deliver them good paying jobs, IMHO. It advises society and parents to not force career choices on their wards which go against their natural interests, and warns against obsession towards engineering career choice. It urges society to advocate excellence (in any field) instead of a particular (engineering) field and opines that excellence is lucrative. [The associated blog post has the exact words.]

...

Pages 38 & 40 carry an excellent article titled, "New age varsities", by Dr. G. Viswanathan, founder and chancellor of Vellore Institute of Technology (the same educational institution whose advertisement on Page 3 is mentioned earlier), <http://www.vit.ac.in/>. I think the entire article is a kind-of must read for those interested in improving technical education in India (or knowing about it).

The associated blog post has three extracts from his article. Given below is a condensed and paraphrased account of those three extracts.

*[Ravi: A deemed university in India is, as per my understanding, an autonomous educational institution recognized by the key national higher education regulators (UGC and/or AICTE) which may have a few campuses. Usually these deemed universities are privately owned and so have to manage their own finances with some limited project grant money from govt. agencies. This is in contrast to government universities which are, I believe, wholly funded by the government. I believe that some of the well recognized deemed universities like Vellore Institute of Technology whose founder is the author of this article, charge substantial amount of fees and other charges (e.g. air-conditioned student hostel rooms with premium charges) from its students. It must also be said that some deemed universities in India have earned a lot of disrepute due to very poor standards of education as well as alleged malpractices.]*

Viswanathan talks of four options for students seeking to enroll in engineering education:

1. IITs (elite institutions with few seats)
2. "Main campus" admission [Ravi: I think that may refer to Anna University's main campuses; TNEA counselling mentioned earlier seems to be conducted by Anna University, which seems to be the dominant and large technical university of Tamil Nadu, and seems to be government funded (a public state university), [http://en.wikipedia.org/wiki/Anna\\_University](http://en.wikipedia.org/wiki/Anna_University)]
3. Colleges affiliated to government university
4. Deemed university

He then quotes a figure of 97 percent of students having to choose between options 3 and 4.

He points out academic advantages of deemed university (note that his university is a deemed university). One of them is the freedom to change, and frequently change, syllabi and curriculum to suit industry needs which he says is difficult to do in government controlled educational institutions. He specifically mentions computing sciences and electronics fields where industry needs change at rapid speed, needing change in syllabi every semester and which, he says, is possible only in deemed universities.

He mentions that industry needs employable graduates who are exposed to practical aspects of the theoretical knowledge (they learn). That needs well-equipped labs (with the lab. work) embedded for the theory subjects. He says that ensuring that students spend quality time in the labs as against the classrooms, helps them be industry-ready.

He mentions that new, innovative industry-oriented programmes can be seen only in deemed universities. He also mentions funding challenges for developmental activities (of labs/courses, one presumes) in government run universities.

But not all deemed universities are good. He says (that the student/parent has) to be careful in selection. He writes, "A candidate has to visit the university, check for the details, talk to the outgoing / passed out students and ascertain the quality before joining."

...

Pages 41 & 42 have an interesting practical advice article, "Choose right", by Dr. Brijesh Nair, Professor and Programme Chair (Civil Engineering) at Vellore Institute of Technology, Vellore, <http://www.vit.ac.in/>. He starts the article by mentioning that many engineering seats remain vacant in South India every year. So if a student wants to pursue engineering no matter what the reputation of the college, getting one of these vacant

seats will be possible. But then he raises the job factor. He goes on to make a rather sweeping statement that if one is looking for just "any job" then any engineering programme in a college with 95 % (or more) campus placement track record should be fine. He also mentions that most students do not have a preference for any branch of engineering but simply want to get a job that pays them well. He writes that the trend observed in campus placements is that students from civil engineering or mechanical engineering (or other fields unrelated to software development) take up software company jobs.

*[I thank The Hindu thenxt.step, and have presumed that they will not have any objections to me sharing the above short extracts from, and many references to, their magazine, in this book, whose ebook is freely downloadable/viewable by all, and whose paperback book is not-for-profit for me (author). This book does not have any financial profit motive whatsoever for the author.]*

## **2014 National (India) Employability report - Poor Employability of Andhra Pradesh Engineering Graduates**

Associated blog post date: 4th Feb. 2014, link: <http://eklavyasai.blogspot.com/2014/02/2014-national-india-employability.html>, short link: <http://bit.ly/2014-nat>

A few days ago, The Hindu had this article, A.P. fares poorly in employability of engineers, <http://www.thehindu.com/news/cities/Hyderabad/ap-fares-poorly-in-employability-of-engineers/article5639970.ece>.

I sent the following comment to the letters email of The Hindu (for the print edition):

I am very happy to see the persistent work done by Aspiring Minds to draw attention to the pathetic employability situation for engineering graduates from Indian, especially South Indian, colleges and universities.

I think its CEO, Himanshu Aggarwal, captures the situation perfectly when he says, "States like A.P. and Tamil Nadu, which have the highest number of engineering colleges, continue to have lowest employability. States need to be conscious towards better education quality rather than building more capacity".

Hopefully the higher education policy makers/regulators and administrators will be able to step in and ensure that suitable career growth incentives are provided to academics who provide good education to students instead of such career growth incentives being focused only on research publications and research projects (with large amount of tax payer grant money). I am not against academic research but the way most academic administrators and regulators seem to focus on research and ignore educating students to become employable is deeply shocking to me. In my humble opinion, the first and foremost duty of an academic should be to teach and teach well - research should be secondary, no matter how much grant money or fame is involved.

--- end comment ---

The above comment did not get accepted for publication in the print edition of The Hindu. However, a shorter version of the above comment (due to the limitation of web page comment size to 1000 characters) on the web page of the article (link given above) was accepted by the moderator and is now shown on it (under my name - Ravi S. Iyer).

Yesterday the "Education Plus" supplement of The Hindu carried an article on similar lines, Engg. graduates lack domain skills, <http://www.thehindu.com/todays-paper/tp-features/tp-educationplus/engg-graduates-lack-domain-skills/article5646438.ece>. I found the following extract in it to be quite significant:

“It is a fact that majority (of) colleges have totally neglected teaching quality. Some blame shortage of good teachers while others genuinely want to hire good teachers but fail to do so due to their non-availability,” agrees N.V. Ramana Rao, Registrar, JNTU Hyderabad.

--- end small extract ---

The Registrar of JNTU Hyderabad (an important technical university of Andhra Pradesh) openly acknowledging that majority of (engineering) colleges have totally neglected teaching quality is a vital acknowledgement of a serious teaching crisis in engineering colleges of Andhra Pradesh. Now who can fix the problem? I don't claim to have all the answers for this tough problem. But a no-brainer suggestion to help solve the problem would be to provide career-growth incentive to those academics who are good teachers instead of focusing only on providing career-growth to those who acquire significant project grant money and publish research papers. For this, the UGC/AICTE regulations for promotion of academics must be changed to introduce some measures of teaching quality, even if they are not perfect, and provide career-growth incentive for those academics who achieve appropriate measure of teaching quality even if they do not have research publications. Let us face it, there is a conflict of interest between teaching and research in Indian academia today, and the poor students face the negative effect of this conflict of interest.

# Miscellaneous

# A Tough View of UK CS Academia

Associated blog post date: 25th Dec. 2011, link: <http://eklavyasai.blogspot.com/2011/12/tough-view-of-us-uk-cs-academia.html> , short link: <http://bit.ly/tough-view>

A correspondent passed on "Hackers and Fighters" by Dr Mark Tarver: [archive link: <https://web.archive.org/web/20111203095605/http://www.lambdassociates.org/blog/hackers.htm> (short link: <http://bit.ly/2HJdhsX> ) for following broken link] <http://www.lambdassociates.org/blog/hackers.htm>.

Dr. Tarver compares the "street programmer"/"street fighter" with the "CS grad."/"school-trained martial artist". I think it is an interesting comparison.

He also talks about how difficult it is to do something innovative in CS academia. He said it took eight years for a CS dept. where he was working to decide to move from Pascal to C++!

He also states that for a fast moving area like computing the university model is too slow to adapt. I entirely agree with his view.

He mentions that the only way of handling the heavy teaching load of five to six courses a year at junior positions in academia is to go for canned courses. I think he is spot on here. At junior positions the teacher himself/herself is struggling to master his/her courses and when he/she is burdened with five to six courses, and fair bit of research work too sometimes, what can the teacher do but go the easy route of canned courses for most if not all the courses he/she has to teach.

He wryly notes that the professor will comprehensively beat the street programmer in the "black arts of churning out papers". I think he is spot on here too :).

Then an acquaintance sent another article by the same author, "Why I am Not a Professor OR The Decline and Fall of the British University". This is a vitriolic article about British universities in general and CS departments of those universities in particular just before 2000. It is brutal in its criticism but offers no thoughts of solutions. Read it only if you can stomach very harsh criticism: [archive link: <https://web.archive.org/web/20111014155153/http://www.lambdassociates.org/blog/decline.htm> (short link: <http://bit.ly/38RxBEr> ) for following broken link] <http://www.lambdassociates.org/blog/decline.htm>. [This seems to be the same article now on a different website: <http://www.marktarver.com/professor.html>.]

I studied the article carefully. My God! What a terrible indictment of the British CS educational system (prior to year 2000)! [I don't mean to imply that India is better off - it may be worse off in the "commoner" universities.] The language used is stinging and sarcastic to the hilt.

The analysis of the need for universities in Europe from the eleventh or twelfth centuries onwards and how free online "information" & general literacy is challenging the position of universities is interesting. So too is the analysis of the impact of British government policies on education post World War II.

I get the feeling that a lot of what the author has penned must be true. But I wonder whether somebody from the British educational system rebutted his article. Maybe this author was with a not-so-great university ["but the procession of students who walked into my office and said 'Dr Tarver, I need to do a final year project but I can't do any programming'... well, they are more than I can remember or even want to remember."]. I find it hard to believe that most British university CS departments would have many students saying, "I need to do a final year project but I can't do any programming". The students, of course, may not be great programmers - but student himself/herself confessing to "can't do any programming" & looking for an easy way out with the project work - that seems hard to believe. Maybe that happens with lots of "commoner" Indian university CS/IT departments too - I just don't know.

The author solved his problem by getting out of the system. I think many of the problems he points out, namely, egalitarianism requirements of politicians who are voicing the needs of the people at large, I guess, not being able to fail many students even if they deserved to be failed as otherwise the course may become unpopular and so be shut down, watering down of courses, farcical (maybe even fixed) teaching audits, the black arts of churning out papers (counterfeit academic Mozarts), academic profession becoming unattractive, foreign immigrant academics with poor language skills ... may be true even today, at least, to some extent of CS education worldwide. But I feel that today the academic system is being challenged by commoner students (the 99 %) (Occupy movement tried to make a point at Harvard), politicians and people at large, and so I am quite hopeful of some meaningful reform happening. The Internet can be a great force which may allow for elite, commoner ... various types of courses to co-exist & flourish based on student choice and industry demand.

On deeper thought, I felt that Dr. Tarver's article is perhaps too negative, paints a depressing picture & offers no thoughts of solutions. A reader may feel that the system is beyond any possibility of redemption. Which I don't think is accurate. What we need to do is to raise awareness of the problems and work amicably and peacefully with politicians, bureaucrats, academics, industry, students & parents to improve the situation.

## The Software Development Professor!

Associated blog post date: 1st Oct. 2011, link: <http://eklavayasai.blogspot.com/2011/10/software-development-professor.html> , short link: <http://bit.ly/software-d>

No, we still don't have such a CS/IT Professor. We have only CS/IT Research Professors. But what if we had CS/IT Software Development Professors? I think such a career track will do wonders for the quality of Design & Code taught in CS/IT academia.

The entry level minimum requirements as per UGC rules are:

3.3.0 The minimum requirements of a good academic record, 55% marks (or an equivalent grade in a point scale wherever grading system is followed) at the master's level and qualifying in the National Eligibility Test (NET), or an accredited test (State Level Eligibility Test - SLET/SET), shall remain for the appointment of Assistant Professors.

Source: [archived link:

[https://web.archive.org/web/20111120222535/http://www.ugc.ac.in/policy/revised\\_finalugcregulationfinal10.pdf](https://web.archive.org/web/20111120222535/http://www.ugc.ac.in/policy/revised_finalugcregulationfinal10.pdf) (short link: <http://bit.ly/39INmxu> ) for following broken link] [http://www.ugc.ac.in/policy/revised\\_finalugcregulationfinal10.pdf](http://www.ugc.ac.in/policy/revised_finalugcregulationfinal10.pdf) (Page 4 of 130).

So a master's degree in CS/IT (M.Tech., MCA etc.) followed by clearing NET/SLET/SET allows a person to be considered for appointment as Assistant Professor (lowest level of university teacher - earlier it used to be Lecturer - I am not including T.A. - Teaching Assistant).

However I get the impression that the NET/SLET/SET exams are very theory oriented. Really don't know how much design & code expertise is tested there. Maybe not tested at all - it is a paper exam.

Alternatively if a person has a PhD in CS/IT he need not clear NET/SLET/SET to be appointed as Assistant Professor.



But the promotion to Associate Professor and Professor is where PhD becomes almost mandatory (Professor - mandatory). Direct recruitment of Associate Professor also seems to have mandatory requirement of PhD. Research publication record is given great value.

Is it any wonder then that CS/IT academicians focus on PhD and research publications? And naturally, design & code in CS/IT academia becomes irrelevant from a career growth point of view. And so, design & code knowledge level among CS/IT teachers in academia is generally poor with students' learning in design & code also being correspondingly poor. This is THE key systemic problem in CS/IT academia.

In a previous post, 'Is a PhD in CS/IT Necessarily a Good Teacher?' (article is included earlier in this book) <http://eklavyasai.blogspot.com/2011/09/is-phd-in-csit-necessarily-good-teacher.html> (short link: <http://bit.ly/is-phd-in>) I had suggested academic teaching CS/IT departments and academic research CS/IT departments. The current AICTE/UGC norms may fit perfectly for academic research CS/IT departments.

But for academic teaching CS/IT departments, different norms based on software development expertise of the teacher may be introduced. Instead of research publication record, we could have Open Source software-development-record. We could also have an impact factor based on how many people are using the software developed.

Norms could be made to decide quality and quantity of software development which can be considered as PhD equivalent. Like the PhD defense, the software development PhD equivalent candidate could be grilled by software development experts (initially from industry but over time from academic software development professors).

Appointment as / Promotion to Associate Professor should recognise Software-Development-PhD-equivalent like current rules recognise Research PhD.

For appointment as / promotion to Software Development Professor appropriate norms of quality & quantity of software development work including guiding/mentoring Software-Development-PhD-equivalent candidates can be arrived at.

So we can have a complete software design & development academic career track with no research component at all for the Software-Development-Professors. Such Software-Development-Professors (Asst. Prof., Assoc. Prof. & Prof.) may be very suitable for academic teaching CS/IT departments. They will be unsuitable for academic research CS/IT departments.

If AICTE/UGC introduce this software-development-professor career track for CS/IT, I think we will see a quantum jump in job-oriented-skill-set quality improvement of CS/IT graduates/post-graduates. This is exactly what most students and their parents want. This is what the CS/IT industry will want. This is what will help the country's economy as industry will not have to spend huge time & money to train 'freshers'. This will allow 'freshers' to straight-away consider becoming software development free-lancers or entrepreneurs (like a Mark Zuckerberg).

Research needs of industry and the country can be met via the academic research CS/IT departments and industry research.

Let supply and demand factors drive how many academic teaching CS/IT departments are set up and how many academic research CS/IT departments are set up. Let us not coax innocent students to do research oriented CS/IT study if what they really want to learn is software development expertise so that they can get a good job. Let students have full and well informed academic freedom about what they want to learn - CS/IT research or software development.

I think this will be a great boon for CS/IT teachers. Right now they are under immense pressure. They have to try to keep some pace with ever changing software technology - so syllabus changes have to be made for courses quite regularly. And they have to produce research publications and take up 'project' work as well.

Then there is an expectation from society that 'computer' professors can help them with software development. All this places a huge and impossible burden on the CS/IT academician.

IMHO, separation into two tracks - Research and Software Development - will simplify life greatly for the CS/IT academician. The Research CS/IT academician focuses on research and teaches research-oriented students. He can safely point software development queries and work to the CS/IT Software Development academician, who focuses on software development and does not bother about research.

Industry software development professionals who would like to contemplate moving to academia can fit in very well in the CS/IT Software-Development-Professor track. Entry of such industry experienced professionals can be a great boon to academic teaching CS/IT departments. AICTE/UGC in conjunction/consultation with software industry experts can come up with norms for deciding equivalence of industry experience quality and quantity wise with academic software-development-record.

As CS/IT is a special field in industry where we have persons from various streams of academia - Physics, Mech. Engg., Metallurgy etc. - the academic degree stream for industry professionals entering into academic teaching CS/IT departments should not be expected to be CS/IT only. If that sounds surprising to CS/IT academic policy makers, Andrew Tanenbaum, the famous CS textbook writer and famous Professor of Computer Science has a PhD in Physics, [http://en.wikipedia.org/wiki/Andrew\\_S.\\_Tanenbaum](http://en.wikipedia.org/wiki/Andrew_S._Tanenbaum). So he moved from Physics to Computer Science. Similar movement happens in CS/IT industry.

These thoughts may seem very unusual. But I think they have significant value if AICTE/UGC policy makers want to fix the serious systemic problems regarding design & code in CS/IT academia that causes significant suffering to countless students and parents in the country.

-----

Given below are two comments from associated blog post.

A fellow industry software techie said in his comment that he has taught theory and programming for M.S. Computer Science students at Pune University, but lacks suitable academic qualifications (B.Sc. is his academic qualification). He said he would love to teach again (in Indian academia) and wanted my thoughts on it. *[To see the exact words of his comment, please visit associated blog post.]*

Ravi S. Iyer (Eklavya Sai Maalik) responded on October 2, 2011 at 10:05 PM:

As I understand the system:

Private Colleges (limited or no govt. funding) - You can be paid "visiting faculty" type - this will be like contractor instead of employee in industry. If what you teach helps in job placement of students, the private colleges may be willing to pay handsomely.

Govt. funded colleges - I think these colleges will be careful not to upset AICTE/UGC as otherwise they may not be given funds! So they may be happy to have you as a FREE "visiting faculty" - no issues whatsoever there about qualifications if they are convinced of your knowledge level. But if you want a paid academician career there, you have to either do a Master's in CS/IT (get above 55%) & clear NET/SLET/SET exam OR acquire a PhD in CS/IT. Otherwise there is no way that you will be taken as a regular "on scale" academician (Asst. Prof.).

BTW if you acquire a PhD then due to your industry experience they may consider appointment as Assoc. Prof.

However my entire understanding above could be wrong. You could check with some engg. colleges.

----

Ravi S. Iyer (Eklavya Sai Maalik) wrote on October 2, 2011 at 10:12 PM:

Oh! I missed out mentioning pre-eminent clause.

AICTE/UGC allow for pre-eminent persons in a field to be made Profs irrespective of their academic qualification. But I guess that would apply to only really well-known experts like a Mr. Narayan Moorthy or Mr. Nandan Nilekani.

And IFIRC 5 journal papers in journals with impact factor of 2.0 and above in the same technical area is considered equivalent to PhD. So if you crack 5 journal papers in your area then you can probably be taken as Assoc. Prof.

Don't know how much value they place on patents. I mean, how many do they expect and how they measure the quality of the patent (impact factor equivalent) to treat it as PhD equivalent.

----

## **Is (Seoul Accord) Accreditation using Outcome Based Assessment methods the way forward to improve teaching standards in Indian CS & IT academia?**

Associated blog post date: 15th June 2014, last updated 16th June 2014, link:

<http://eklavyasai.blogspot.com/2014/06/is-accreditation-using-outcome-based.html> , short link: <http://bit.ly/is-accred>

It seems that the top technical education academic administrators in the country involving Ministry of Human Resource Development (MHRD), <https://mhrd.gov.in/>, All India Council for Technical Education (AICTE), <https://www.aicte-india.org/>, and the National Board of Accreditation (NBA), <https://www.nbaind.org/>, are taking steps in the direction of Outcome Based Education being adopted by technical education institutions in the country. This applies to all technical education in which Computer Science and Information Technology are two streams (besides streams like Civil Engineering, Electrical Engineering etc.).

Here's a report in The Hindu today, 'NBA accreditation helps students secure quality education', <http://www.thehindu.com/todays-paper/tp-national/tp-andhrapradesh/nba-accreditation-helps-students-secure-quality-education/article6116366.ece>, about such efforts in the Andhra Pradesh district of Chittoor. Two small extracts:

"A three-day National Board Accreditation (NBA) workshop on 'Outcome-based Accreditation', under the aegis of Jawaharlal Nehru Technological University – Anantapur's (JNTUA) NBA Nodal Centre, was inaugurated at Sri Venkateswara College of Engineering (SVCE) here on Saturday." [Ravi: Here refers to Tirupathi, a city in Chittoor district of Andhra Pradesh.]

...

"They explained about the existing model of NBA i.e., 'Outcome-based Accreditation' and spoke on its process, validation and several other issues like how to define vision and mission of a programme, how to map Program Outcome (POs), Course Outcomes (COs) and Program Enabled Outcomes (PEOs) supplemented by its benefits to technical institutes."

--- end extracts ---

From <https://www.nbaind.org/about>, "NBA came into existence as an independent autonomous body with effect from 7th January 2010 with the objectives of assurance of quality and relevance to technical education, especially of the programs in professional and technical disciplines, i.e., Engineering and Technology, Management, Architecture, Pharmacy and Hotel Management and Catering Technology, through the mechanism of accreditation of programs offered by technical institutions." ... "Over the period of its existence, the NBA has introduced a new processes, parameters and criteria for accreditation that are in line with the best international practices and oriented to assess the outcomes of the programme."

Ravi: So the stamp of approval from the top level in the country for Computer Science and Information Technology academic programs (e.g. B.Tech. Computer Science & Engineering, M.Tech. Computer Science & Engineering) is the NBA accreditation. NAAC may not be the suitable accreditation organization for CS & IT programs.

Outcome Based Education (OBE) seems to be the big thing for NBA. The two Powerpoint slides' files here: [archived link: <https://web.archive.org/web/20140331000410/http://www.nbaind.org/En/1055-learning-resources.aspx> (short link: <http://bit.ly/3c1RnPW> ) for following broken link] <http://www.nbaind.org/En/1055-learning-resources.aspx> give some idea of NBA's approach.

I have given below some key content related to OBE from one of its two Powerpoint files (Overview of Outcome Based Accreditation and Outcome Based Education – [archived link: (password not needed to open as read only) [https://web.archive.org/web/20140701000000\\*/http://www.nbaind.org/files/oba\\_nba2.pptx](https://web.archive.org/web/20140701000000*/http://www.nbaind.org/files/oba_nba2.pptx) (short link: <http://bit.ly/2SPiHsT> ) for following broken link] [http://www.nbaind.org/files/oba\\_nba2.pptx](http://www.nbaind.org/files/oba_nba2.pptx) given in the above link:

[From slide 6]

What are the Outcomes?

Outcomes with reference to Educational Programme are the competencies, skills, knowledge and proficiency a student is expected to get at the time of graduation out of the programme.

Outcomes in educational terms could be Program Outcomes and Course Outcomes.

Program outcomes are statements that describe what students are expected to know and be able to do by the time of graduation.

...

[From slide 19]

Outcome Based Education

Starting with a clear picture of what is important for students to be able to do ...

Then organizing the curriculum, delivery and assessment to make sure learning happens ...

...

[From slide 20]

Outcome Based Education -> Outcome Based Curriculum (What the student should be able to do?) -> Outcome Based Learning & Teaching (How to make the student achieve the outcome?) -> Outcome Based Assessment (How to measure what the student has achieved?)

--- end slide extracts ---

Ravi: The above extracts seem to be really great on the face of it. But the wikipedia page for Outcome Based Education, [https://en.wikipedia.org/wiki/Outcome-based\\_education](https://en.wikipedia.org/wiki/Outcome-based_education), gives a mixed account. It seems that OBE did not go well in some places, and it seems to be not so popular a term in the USA now.

The Washington Accord uses OBE. First about the Washington Accord from [http://en.wikipedia.org/wiki/Washington\\_Accord](http://en.wikipedia.org/wiki/Washington_Accord), "The Washington Accord is an international accreditation agreement for professional engineering academic degrees, between the bodies responsible for accreditation in its signatory countries. Established in 1989, the signatories as of 2014 are Australia, Canada, Chinese Taipei, Hong Kong China, India, Ireland, Japan, Korea, Malaysia, New Zealand, Russia, Singapore, South Africa, Turkey, the United Kingdom and the United States."

Additionally, "The Washington Accord covers undergraduate engineering degrees under Outcome-based education approach". The wiki page links to this pdf explaining OBE - <http://www.utar.edu.my/fes/file/OBE.pdf>.

Here's an interesting recent Indian news item "Boost for engineers: India now part of Washington Accord", <http://timesofindia.indiatimes.com/home/education/news/Boost-for-engineers-India-now-part-of-Washington-Accord/articleshow/36519279.cms>.

Some notes and comments:

India has become the 17th member of the Washington Accord after a lot of effort including some failures.

...

Washington Accord seems to exclude CS & IT which are covered by Seoul Accord.

...

"Becoming part of Washington Accord also does not necessarily mean that all engineering degrees by all Indian colleges will get equivalence with those of other member countries. NBA has shortlisted 220-odd engineering colleges as Tier-I institutes whose undergraduate engineering programme is in tune with what is required under the Accord."

*[Ravi: This effort by NBA sounds encouraging. IT is excluded - so that is not good news. Hopefully NBA and MHRD are working on becoming a member of the Seoul accord too. BTW I did not know of these Washington and Seoul accords till I read news reports about the Washington accord recently.]*

...

Extensive verification of these Tier-I institutes will need to be done prior to them being deemed as Washington Accord educational institutions. It is expected that courses will be redesigned to focus on outcomes and students given freedom to explore and innovate.

*[Ravi: Excellent! I love the emphasis on outcomes (presumably involving a large component of learning outcomes). This kind of initiative will force engg. colleges to reorient their thinking towards improving the learning outcomes of their undergraduate programs.]*

...

Other educational institutions have been given a roadmap by NBA to be followed if they want to attempt becoming institutions covered/approved by Washington Accord.

*[Ravi: Sounds good.]*

---- end notes and comments ---

Here is a link with Washington accord listed member countries (India is still shown as provisional status member), [archive link:

<https://web.archive.org/web/20140218094428/http://www.ieagreements.org/Washington-Accord/signatories.cfm> (short link: <http://bit.ly/38SJ0UD> ) for following broken link]

<http://www.ieagreements.org/Washington-Accord/signatories.cfm>.

Here's another article about India becoming a signatory to Washington Accord:

<http://www.livemint.com/Politics/Rf8g7vPOsU2A9cZmpYnHMN/India-gets-permanent-membership-of-Washington-Accord.html>.

*[Ravi: The above link seems to give inside info. about how it was pushed through. It seems that former HRD ministers Shri Kapil Sibal and Shri Pallam Raju deserve congratulations for having pushed this through from a political and administrative will point of view. Separation of NBA from AICTE may have been the real big decision that paid dividends. NBA chief and former IIT Delhi Director Prof. Surendra Prasad, Dr. D.K. Paliwal and Shri Ashok Thakur of MHRD are reported to have been key contributors for this effort and deserve to be congratulated (perhaps congratulated more as they may have been doing the real hard work).]*

Ravi: Now about the Seoul Accord as that is what comes into play for CS & IT field. I struggled a little to get quick overview info. of the Seoul Accord. Here's what I have been able to garner:

From [archive link: <https://web.archive.org/web/20140402210458/http://www.abet.org/computing-mra-seoul-accord/> (short link: <http://bit.ly/37L2okZ> ) for following broken link] <http://www.abet.org/computing-mra-seoul-accord/>:

The Seoul Accord, established in 2008, is a mutual recognition agreement pertaining to computing and IT-related programs accredited by its signatories within their respective jurisdiction. Signatories to the Seoul Accord are organizations responsible for accrediting computing and IT-related programs in Australia, Canada, Chinese Taipei, Hong Kong, Japan, Korea, the United Kingdom, and the United States.

*[Ravi: That's pretty recent - around 6 years ago. No wonder it is not so well known.]*

...

Graduate attributes form a set of individually-assessable outcomes that are indicative of a graduate's potential competency. The graduate attributes are exemplars of the attributes expected of a graduate from an accredited program.

The Seoul Accord has identified the characteristics of graduates of all computing programs that fall within the scope of the accord. A signatory may identify additional attributes that differentiate specific programs accredited by the signatory.

--- end extracts from abet.org ---

Details of the graduate attributes are available at the Seoul accord website here: [archive link:

[https://web.archive.org/web/20150314052729/http://www.seoulaccord.com/accord/contents.jsp?menu\\_l=144&menu\\_m=195&menu\\_s=236](https://web.archive.org/web/20150314052729/http://www.seoulaccord.com/accord/contents.jsp?menu_l=144&menu_m=195&menu_s=236) (short link: <http://bit.ly/2PgL9lh> ) for following broken link]

[http://www.seoulaccord.com/accord/contents.jsp?menu\\_l=144&menu\\_m=195&menu\\_s=236](http://www.seoulaccord.com/accord/contents.jsp?menu_l=144&menu_m=195&menu_s=236) but it is not an easy first read.

Ravi: Perhaps one can say that the graduate attributes of the Seoul accord are roughly the outcomes expected, and so the Seoul Accord could implicitly promote Outcome Based Education.

I guess NBA would be trying/will try to get on board with the Seoul Accord too. Once that gets done then Indian Computer Science and Information Technology departments of technical education institutions can

strive for Seoul Accord accreditation through NBA. That may lead to emphasis on well defined learning outcomes of CS & IT graduates and post-graduates in India. These learning outcomes would also include, I presume, the practice of software development.

So, it seems to me, the Seoul Accord may be a great thing for NBA and then Indian CS & IT academic departments to get on board from a perspective of improvement in knowledge of and skill in the practice of software development of Indian CS & IT graduates & post-graduates. Here's the link for NBA's manual for accreditation of UG engineering programs, first edition, dated March 2012, 93 pages:

<http://www.nbaind.org/Files/engineering-programs.pdf>. But in my quick browsing of this doc. and of the NBA site, I did not get well defined course specific outcomes for courses in CS & IT (e.g. C++ programming or Network programming lab. courses or even more generic courses like Operating Systems).

ABET seems to be a big accreditation organization in the USA (and other countries). Interestingly it has a find accredited programs link: <http://main.abet.org/aps/Accreditedprogramsearch.aspx>. Computer Engineering, Computer Engineering Technology and Computer Science are three separate listed programs. Similarly Information Engineering Technology, Information Systems and Information Technology are three separate listed programs.

Searching for Computer Science in India lists only VIT university in its results. [22<sup>nd</sup> Feb. 2020 Update: Searching today with same parameters on above link (which is redirected to another link automatically) lists 4 (educational) institutions.] But its CS entry does not show Seoul Accord status, as expected (since India (NBA) is not yet a signatory to Seoul Accord).

Searching for Computer Science in USA lists 265 results! [22<sup>nd</sup> Feb. 2020 Update: Searching today with same parameters lists 345 (educational) institutions.] I checked out a few for Seoul Accord status for its CS program. All the ones I checked - Arizona State University, Florida State University, Georgia Institute of Technology, Massachusetts Institute of Technology, Michigan State University, Rochester Institute of Technology, State University of New York at Binghamton, Texas A&M University, University of California, Berkeley, University of California, Riverside, University of Massachusetts Boston, University of New Hampshire, University of Virginia and Washington State University - have Seoul Accord status for its Computer Science program(s)! [Some have Computer Engineering programs having Washington Accord status not Seoul Accord! But I will skip that detail for now.]

Clearly the way forward for Indian CS & IT academic departments is to get their programs under Seoul Accord. For that, first MHRD and NBA (and AICTE implicitly) have to engage with Seoul Accord folks like they engaged with Washington Accord folks, and do the needful, even if it involves lots of hard work and some failures, for India (NBA) to become a signatory to the Seoul Accord.

To conclude this post, let me return to the title question, is (Seoul Accord) accreditation using Outcome Based Assessment methods the way forward to improve teaching standards in Indian CS & IT academia? I think the answer seems to be yes. In any case, MHRD, AICTE and NBA are going the (Washington Accord) accreditation using Outcome Based Assessment methods way from an entire technical education perspective. So, it seems to me, technical education institutions of India do not have much of a choice and will have to follow Outcome Based Education and then get accredited by NBA.

I hope that the learning outcomes for CS & IT field in NBA accreditation, in terms of both program outcomes as well as (individual) course outcomes are defined well. In particular, I hope that the learning outcomes for software lab. courses are defined well.

*[I thank thehindu.com, nbaind.org, wikipedia.org, indiatimes.com and abet.org, and have presumed that they will not have any objections to me sharing the above short extracts from, and many references to, their website/articles, in this book, whose ebook is freely downloadable/viewable by all, and whose paperback book is not-for-profit for me (author). This book does not have any financial profit motive whatsoever for the author.]*

# US CS PhD student's Comparative View of US and Indian Academia

Associated blog post date: 7th June 2013, last updated 8th June 2013, link:

<http://eklavyasai.blogspot.com/2013/06/us-cs-phd-students-comparitive-view-of.html> , short link:  
<http://bit.ly/us-cs-phd>

I had an interesting mail exchange about US and Indian academia with a correspondent who is doing his PhD in Computer Science at a US university and also acts as a TA for programming courses there. I thought readers may be interested in the exchange and so have given an edited version of the exchange below. The PhD student's views are based on his limited exposure and so may be "off target" at times. However, I think it is a fresh view of a relative newcomer to US CS academia and so is a valuable one. People who are in a system for many years tend to get used to the system and accept it without questioning it much.

The correspondent (referred later as Corr) wrote: There is however one thing I would like to mention. You wrote in your email, "Such rewards and punishments will ensure that teachers will develop the right attitude towards their primary duty of teaching else they will be either left behind in their career or even asked to pursue another career option." [Ravi: Note this refers to a previous mail exchange between me and the correspondent on student feedback being used as a measure for teaching quality. Most of that exchange is captured as part of the post (given earlier in this book), Discussion on Concrete Suggestions for Measuring Teaching Quality ..., <http://eklavyasai.blogspot.com/2013/06/discussion-on-concrete-suggestions-for.html> , short link: <http://bit.ly/disc-conc>.

I would like to bring to your notice that here in the US, teaching is not the primary duty for the instructor employed at an University. There are 3 responsibilities for a professor and they have roughly equal weight. 1) Teaching, 2) Research and 3) Service.

Ravi responded: In previous interactions I had with another person who is also doing a PhD in USA, I was informed that in the US, broadly there are higher-ranked research universities and lower-ranked teaching universities. Most or all of the lower-ranked universities want to become higher-ranked research universities.

A related extract & link from one of my blog posts, <http://eklavyasai.blogspot.in/2011/12/affordable-subject-wise-certification.html>, is given below:

US President Obama meets US university presidents to address/discuss their challenges:  
<http://www.economist.com/node/21541398>.

The article talks of the problems of rising costs in an age of austerity, more courses & more research students than there is money for and interestingly, Ivy league envy. "Ivy League envy leads to an obsession with research" it states. This results in professors who are focused on research and don't do their job of teaching students well enough, and even causes teaching dysfunction at lower-level universities!

I think the last problem is the case with lots of Indian universities too.

--- end blog post extract ---

I think what you have written above - roughly equal weight to teaching, research and service - may apply only to US research universities and not to US teaching (intensive) universities.

Corr: **Teaching:** Undertaking to teach a course at the University. Usually a professor does not teach more than one course a semester and may sometimes teach alternate semesters too.



Ravi: Interesting. I was given the impression it is similar in IITs.

Corr: There are, however, lecturers who teach 2-3 courses a semester, but they do not have the burden of Research or Service.

Ravi: Oh! So then this becomes a teaching track option. I was told by a senior Indian academic that IITs used to give that option to its faculty - teaching track with no research, or research track with low teaching load. And for a teaching track academic even in a research university, teaching will be the primary duty!

Corr: At the graduate level the classes may have 10-70 students and the undergraduate level may have 100-800 students.

Ravi: 800! Mannn! I thought it would go up to maybe 150 or so. Are you sure about 800? Have you seen any such UG class with around 800 students?

Corr: The instructor has a proportional number of TAs to help with grading and course administration.

Ravi: Interesting point - the proportional number of TAs makes sense.

*[Later exchange. Corr: Yes, classroom sizes do get that large. I have myself been a TA for a class size of 250. The number 800, I came across from a freshman at UC Berkeley.]*

*Ravi: I see - very interesting info. As I thought more about it, so long as the classroom facilities are good with all students being able to see the presentations part and hear the lecturer clearly, I now feel it may be something that works out decently. For the hands-on lab. part, as you wrote earlier, there would be proportional number of TAs for the large number of students.*

*If the classroom size is reduced to some fixed lower number, then interested students do not get a chance to learn the course! I had read that in Stanford the majority of the students who take the Introduction to Programming course are non CS majors! I think programming has become an important requirement for a wide spectrum of fields apart from Computer Science/Information Technology. So, I can now imagine 800 students for a programming course in a large US university. Hmm. My imagination did need some stretching to handle that though :).]*

Corr: **Research:** The professor is expected to publish his research work in Conferences and Journals applicable to his/her domain and bring visibility to the University. The adage "Publish or Perish" still holds in many Universities.

Ravi: I was told by a senior US academic that it applies particularly to those on tenure-track. Without a good publication record tenure-ship will not be granted.

Corr: Depending on the nature of the work, some Universities would like their departments to encourage industry initiatives/collaborations and enhance the University visibility through media interaction. In such places the motto changes to, "Demo or Doom".

Ravi: That's a new one for me :). Interesting!

Corr: The most important aspect of research is to write grants where the professors seek funding from a multitude of agencies to support themselves and their students in doing research.

Ravi: Well, that seems to be a slight exaggeration :). Of course, the research grant money is vital but research excellence in terms of research approach, results and publications with visibility/impact would be critical, wouldn't it?

Corr: **Service:** As an active member in the publishing community, the professor must be available during the year to review publications that have been submitted to conferences and journals in his/her domain.

Ravi: I think the prof. must be choosing a few publications/conferences for this service and turning down others.

Corr: He/she must also be ready to chair a session in a conference, organize a workshop etc. as part of a bigger conference. (All the major conferences are administrated by professors across the world).

Ravi: Once again they must be choosing a few.

Corr: They must also assist students by being in their committees and guide their thesis(Masters/PhD). A gamut of activities are included in service with no remuneration being involved.

Ravi: Well, I think there would be some work load distribution norms with time periods allotted to different types of work. PhD students being guided clearly would be part of their work load as seen by academic administration - that's how it is here. For the others a certain level of activity may be expected and appropriate ("free") time given.

In India, the impression I get is that as one becomes a senior Prof. the teaching load can go down even to zero, and the Prof. then is given enough time to play all these roles, on his/her full Professor salary. So, to me, it is not as if the Professors do the 'Service' activities on weekends and spare time after work. They must be doing the bulk of it within their regular worktime.

Corr: It is therefore unfair to judge a professor only based on his/her teaching and student feedback. *[Ravi: Note this refers to previous mail exchange(s) between me and the correspondent on student feedback being used as a measure for teaching quality. Most of that exchange is captured as part of the post mentioned earlier in this article: Discussion on Concrete Suggestions for Measuring Teaching Quality ...]*

Ravi: I certainly was not suggesting that. I was limiting myself to measuring (judging) teaching quality of the teacher role played by the academic. I went even further by limiting myself only to teaching of lab. courses./practice of software development. Other roles that the academic plays have to be measured/judged differently and I am not touching upon that at all.

Corr: There is also another reason behind this aura of untouchability that professors have. It is my understanding that once a professor is tenured, in most cases, there is nothing the department/University can do to dislodge them (unless in very exceptional cases).

Ravi: I think you are right. I had read up about it some time back and it seems the reason (or one of the main reasons) is to provide the professor the freedom to explore new lines of thought/knowledge against popular opposition.

Corr: Coming to India: Most of this is directly applicable to the big players like IISc and to some extent the IITs in India. But, a majority of the Engineering institutions have professors only doing teaching for most of the year. There is definitely Research to take into account, but the major difference between research in India and the US is this IMHO: students in India are funded by UGC and professors do not have that burden of funding students. (This is a big big bonus for professors).

Ravi: Interesting! I did not realize/know that US does not fund research students via a UGC equivalent setup. In India, I am given to understand that UGC/CSIR pays research fellowship money (for max. 5 years) only for those students who clear the UGC/CSIR JRF NET exam (Junior Research Fellowship, National Entrance Test). I think there must be significant number of research scholars in UGC/AICTE educational institutions who do not clear JRF NET exam. They may have to be supported with some (usually lesser) stipend by the university itself or through some project grant funds.

Corr: Professors still have to write grants to get funding for research (equipment/resources), but that is not a big problem. They would not lose their job if none of their grants got funded.

Ravi: That seems to be correct.

Corr: In fact, I have heard from reliable sources that the UGC funding for research grants in India is never completely used up. (This could be because of lack of worthy proposals or mere bureaucracy - and that is just speculation). This has its pros and cons. As a con, IMHO, this could be one of the reasons for the lackadaisical attitude towards research in India. This is also one of the reasons for the reverse brain drain in the academic community. Its definitely an easy life in India for the professors.

Ravi: I see - reverse brain drain is interesting. But perhaps it has been triggered by US (and world) economic crisis of 2007-08, which, I believe, is still not fully resolved, resulting in major cuts to government funding of US academic stuff (research, teaching ..).

Corr: In the US, the professors fund their PhD students doing research under them. There is a lot of pressure to write grants and get funding.

Ravi: Yes, I had read a rather harrowing article on it, recently. In case you want to have a quick look, here's a blog post which has a Forbes article link about current US researcher issues and my comments on it, (article is given earlier in this book) Is Academic Research Grant Money Corrupting Academic Teaching Ideals?, <http://eklavyasai.blogspot.com/2013/04/is-academic-research-grant-money.html> , short link: <http://bit.ly/is-academic>.

Corr: Also, the student teacher ratio is highly unfair to the teachers in India. To my knowledge, there is no concept of TAs in most Universities across India and the professors have to teach more than one course in a semester. It is still a huge burden.

Ravi: I think in India, for lab. courses (in UGC/AICTE institutions) which have large number of students, there must be some support for the main teacher. But I don't know the exact situation.

Corr: Then, there is the service aspect, but most of the major international conferences are headed by US/European/Australian faculty. There is representation from India mainly from the IISc and the IITs. Service is not a major factor for the Indian professor, IMHO.

Ravi: Nowadays there is a lot of pressure to publish for UGC/AICTE academics too. And there are lots of Indian conferences and even journals. So, besides IITs and IISc, there certainly is significant research publication work that UGC/AICTE Indian academics are doing.

Corr: And then there is bureaucracy, and I do not have time for that.

Ravi: :)

Corr: In short, research in India is a completely different animal as compared to research in the US. It would not be fair to draw exacting parallels between the two.

Ravi: I entirely agree. One factor you don't seem to have brought out is the massive endowment that top research universities in USA have. MIT and Harvard have some super-duper endowment - that gives them the financial muscle to recruit top notch academics and have light teaching load for them. In India, I think equivalent financial muscle is available only to few institutions like IITs and IISc.

Dug up some data: According to a USA News report, <https://www.usnews.com/education/best-colleges/the-short-list-college/articles/2013/10/01/universities-with-the-largest-financial-endowments-colleges-with-the-largest-financial-endowments>, for fiscal 2011 Harvard had an endowment of US \$ 32 billion, Yale 19 billion, Princeton 17 billion, Stanford 16 billion and MIT 9 billion. In rupee terms (1 US \$ = Rs. 57) Harvard's endowment is Rs. 1,82,472 crores i.e. One Lakh Eighty Two Thousand Four Hundred and Seventy Two crores! [Calc. (US \$) 32,012,729,000 x 57 =

(Rs.) 1,824,725,553,000 ]. MIT's endowment is Rs. 55,361 crores [Calc. (US \$) 9,712,628,000 x 57 = (Rs.) 553,619,796,000 ].

I tried getting endowment fund figures for IITs in India but it is not easily available. Here is some info., [archive link: <https://web.archive.org/web/20130530200452/http://www.iitbombay.org/giving-back/how-to-donate-india> (short link: <http://bit.ly/2HMg4Sn> ) for following broken link] <http://www.iitbombay.org/giving-back/how-to-donate-india>, which states, "The top 10 US universities in the US have endowments in excess of \$ 100 billion. In comparison IIT Bombay's alumni donations raised Rs 15 crores last year." [Rs. 15 crores = Rs. 150 million which at Rs. 57 for 1 US \$ converts to US \$ 2.6 million. Calc. 150,000,000 / 57 = 2,631,579 (rounded).] How much the government is contributing to IIT Bombay was not easily available. I wouldn't be surprised if the govt. contribution figure is pretty big by Indian standards - hundreds of crores (rupees) or more.

My view based on information gathered over the past year or two is that, in both the US and India, a research university has to have powerful financial backing. If an educational institution does not have much financial muscle it can survive only as a mainly teaching university/college dependent on tuition fees from students as a major source of income with which to meet its expenses. Some small research (publication) work could be there with financial grant support from appropriate government departments/organizations (in India it would be UGC, DST etc.). [Small free education institutions supported by a small corpus fund are a special case. IMHO, they too can mainly function as a teaching university/college with some small amount of research (publication) work. If they want to be very ambitious on research side and compete with top research universities (IITs/IISc. for Indian context), they first need to ensure huge endowment/corpus funds.]

Corr: But I do admire your efforts in reaching out to the authorities to usher in some drastic changes that would improve teaching in India. I believe it does need a revamp.

Ravi: Thanks for the kind words. I am trying to do my bit with the focus being improving teaching of the practice of software development in UGC/AICTE regulated Indian CS/IT academia. That's an area where I believe I now have enough knowledge to comment rather authoritatively. I steer clear of other areas - even CS/IT theory course teaching or elite IIT teaching.

## **Prof. Stroustrup: Software development is a potentially noble profession like medicine or classical engg. disciplines but long way to go**

Associated blog post date: 26th Aug. 2014, link: <http://eklavyasai.blogspot.com/2014/08/prof-stroustrup-software-development-is.html> , short link: <http://bit.ly/prof-stro>

Given below is the link of a very interesting interview of one of the living legends of the software development field, Prof. Bjarne Stroustrup, creator of C++, <http://www.stroustrup.com/>, [http://en.wikipedia.org/wiki/Bjarne\\_Stroustrup](http://en.wikipedia.org/wiki/Bjarne_Stroustrup). I found this statement of Stroustrup (in the interview) to be quite insightful as well as visionary, "I see software development as a potentially noble profession, like medicine or some of the classical engineering disciplines, but we still have a long way to go to get there."

The article is dated Dec. 2013, 'We need better balance between theory and practice', Bjarne Stroustrup, Father of C++. <http://yourstory.com/2013/12/bjarne-stroustrup-interview/#> (short link: <http://bit.ly/38TvSie>).

I think it will be really great if software development practitioners view their profession with the maturity and responsibility that medical practitioners & engineering professionals view their professions. Of course, there are some bad apples among medical practitioners & engg. professionals too, especially in economically developing countries including India, but overall they are a far more accountable & responsible set of people, with that accountability being mandated by law in many cases.

# Appendices

# Appendix A: My software field biodata, software industry work experience and lab. courses I taught

## Software Field Biodata of Ravi S. Iyer

Last Updated On: 23rd Feb. 2020

### Current Role

No longer involved in software development or teaching related work except for converting software related past blogs into (blog)books.

### Experience

18 years in software development industry (in U.S., Europe, Far East Asia and India) playing roles right from programmer, designer, project manager, software development manager to finally a software consultant role. Have also been a trainer teaching programming & design in industry; Retired from commercial work in Aug. 2002.

Subsequently, as free Seva (free service), for around 9 years, have taught programming (lab) courses in a deemed University in India at M.Tech. (CS) and M.Sc. (Maths & CS) level and been a technical and programming consultant for M.Tech. (CS) projects and M.Sc. (Maths & CS) dissertation computer projects.

Mainly from 2011 to 2014, have worked on three software related blogs:

- Peaceful and amicable, Indian Computer Science (CS) & Information Technology (IT) academic reform activism: <http://eklavayasai.blogspot.com/p/table-of-contents.html> (short link: <http://bit.ly/eklavya-toc> ).
- Ravi S. Iyer's Software Lab. Courses: <https://raviiyerteaches.wordpress.com/> (short link: <http://bit.ly/ri-teaches> ). It has the content of software lab. courses taught by me while offering free service with designation of Honorary Staff/Honorary Faculty/Visiting Faculty to a Mathematics & Computer Science department in a deemed university in Andhra Pradesh, India, from 2003 to 2011.
- "Service to Society" Free and Open Source IT software development especially aimed at rural Indian society: <http://raviiyer.org> (mapped to <https://ravisiyer.wordpress.com/> ). This work did not really take off due to lack of volunteers. However, it has some free software licensing (FOSS Licensing) related posts which may be useful to some readers. Now I use the blog to record user-level tech. matters (e.g. Mobile Internet speeds in Puttaparthi where I now live).

### Technical Areas worked on while in Software Industry

Architecture, Design, Coding and/or teaching/mentoring roles in/using:

- Web technologies using C#/ASP.Net and Java 2 Enterprise Ed.
- OOAD incl. Design Patterns and UML
- GUI Design
- TCP/IP socket prog, VoIP MS API
- SMTP gateway, X.400, X.500, Sendmail
- Banyan Vines Network OS, RPC
- Videotex
- IIS & Netscape web server API
- Web Services (SOAP)
- Document Imaging Indexing systems

- C/C++, STL, Visual C++/MFC, Visual Basic, Oracle, Sql Server, Informix, PowerBuilder,
- SQLWindows, X-Windows API, Wang VS Assembler, COBOL
- Microsoft Windows, Unix/Linux/Solaris and Wang VS operating systems.
- Misc: ATL/COM, Multi-threading, XML, Windows Hooks

### **Academic Teaching Experience Details**

Have taught (and introduced most of) the following (lab.) courses at M.Tech., M.Sc. & B.Sc. levels in a Mathematics & Computer Science department of a deemed university in Andhra Pradesh, India:

- Open Source Web Technology (As applied to a free school educational web portal)
- Advanced Unix Programming
- Network Programming (TCP/IP Sockets)
- Multi-Threaded Programming
- Minix/Linux Kernel Internals
- C++ Programming
- C Programming
- Java Web Programming
- C# & ASP.Net Web Programming
- VC++/MFC/Windows Programming

### **Technical & Programming Consultancy in Academia**

Was technical and programming consultant for M.Tech. (CS) projects in the following technical areas:

- Ontology for Indian school education
- Re-engineering & Object Oriented Analysis & Design (OOAD)
- Web Services Security & Security of Coalition of Web Services
- File systems in Linux/Minix
- VC++ programming, detecting & debugging memory leaks

Was technical and programming consultant for M.Sc. (Maths & CS) Dissertation computer projects involving the following technical & functional areas.

Technical areas:

- ASP.Net Web programming
- Database design and programming
- Graphical User Interface (GUI) design

Functional areas:

- E-learning of conversational English from Telugu & vice-versa.
- Agricultural Portal
- Department research publications database
- Hostel Railway Reservation



## Academic Publication Record

Have published the following academic papers as a co-author:

1. A.Sateesh, R.Iyer, G.V.Prabhakar Rao, S.Kapoor and R.Mukkamala, "SPN: A Web Services Security Policy Negotiator", International Conference on Web Engineering and Applications (ICWA), Bhubaneswar, pp 137-148, December 2007.
2. S.Sampath, R.Iyer, K.S.Sridharan, R.Mukkamala and S.Kapoor, "Secure Web-based Sharing of Health Information Services Using Ad-hoc Dynamic Coalitions", International Conference on Information Technology : New Generations (ITNG), Las Vegas, pp 297-302, April 2008.  
[http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=4492495](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4492495).
3. S.Sampath, R.Iyer, K.S.Sridharan, R.Mukkamala, S.Kapoor, "Coalition Service Registry for Ad-hoc Dynamic Coalitions", Enterprise Information Systems and Web Technologies, pp 41-46, 2008.

Following is my single-author paper, a preprint, on arxiv.org:

1. Ravi S. Iyer, "Improve the Practice of Software Development in India by Having a Software Development Career Track in Indian CS & IT Academia", arxiv.org preprint, last revised 21 Dec 2012. Preprint Abstract: <http://arxiv.org/abs/1202.1715>; Preprint Pdf: <http://arxiv.org/pdf/1202.1715>.

## Academic Qualification

B.Sc. (Physics) in 1983.

Dropped out of M.Sc. (Physics) due to financial problems. That turned out to be a blessing in disguise as I was able to enter the software industry at a younger age ☺.

## Age

57 years

---

## Industry Work Experience of Ravi S. Iyer

### Industry Experience Overview

18 years in software development industry (in U.S., Europe, Far East Asia and India) playing roles right from programmer, designer, project manager, software development manager to finally a software consultant role. Have also been a trainer teaching programming & design in industry; Retired from commercial work in Aug. 2002.

### Employment/Consultancy History

Jul 2001 to Aug 2002

Technical Consultant with CelPro (a Dutch startup), SEEPZ, Mumbai.

Nov 1993 to Jun 2001

Technical Consultant with Mastek, SEEPZ, Mumbai.

Mar 1993 to Oct 1993

Freelance Corporate Training Instructor in Mumbai.

Jun 1990 to Nov 1992

Employed as Software Manager with Boshu Technics (a startup), SEEPZ, Mumbai.

Mar 1984 to May 1990

Employed with Datamatics Consultants, SEEPZ, Mumbai; Started as Trainee Programmer and last position with Datamatics was Project Leader.

### **Major Projects (on-site and off-site)**

#### **Product Development (Description: Technical Areas Involved)**

- Application Framework on .Net to easily build database forms oriented web applications, for Dutch customer: Architecture, OOAD, ASP.Net, C#
- Internet User Profiling Product for EngageTech, USA: C++, TCP/IP, Multi-threading, ISAPI, NSAPI, CGI, Windows NT, Sun Solaris, IIS & Netscape web servers, ODBC, Internationalization
- Visual Interface for Object Oriented Business applications tool for Mastek, India: GUI Design, OOAD, Visual C++
- Automated Tester for Computer Telephony Interface product for Japanese customer: C++, Windows, Multithreading, Windows Hooks
- Document Imaging Indexing product for Discorp, USA: Windows, C, Oracle, Imaging software
- SMTP E-Mail Gateway for Wang Laboratories, USA: Unix, X.400, X.500, Sendmail, TCP/IP, OPEN/OFFICE, C
- Port of Banyan Vines PC LAN OS Backup utility from Unix to Wang VS OS for Wang Laboratories, USA: Wang VS OS, Unix, Banyan Vines, RPC, C
- Response module of Wang VS Videotex software for Wang International Telecommunications Research Centre, Belgium: Wang VS OS, VS Videotex, VS Assembler, C
- Wang VS Operating System Services Quality Assurance for Wang Laboratories, USA: Wang VS OS, OS Dump Analysis Tools, VS Assembler, C
- Wang Word Processor (WP/WP+) Archiving Utilities for Wang Laboratories, USA: Wang VS OS, VS Assembler
- Window Manager study for Baan Info Systems, Netherlands: X-Windows, Microsoft Windows, UNIX, SunOS
- Executive Information System Product for Boshu Technics, India: X-Windows/Motif, C, Informix

#### **Custom Software Development (Description: Technical Areas Involved)**

- Service Management application for Dutch customer: OOAD, ASP.Net, C#
- Process Simulator for German customer: Architecture, OOAD, C++ & Multithreading
- ActiveX Graph Controls for Wireless Devices Monitoring for Japanese customer: Visual C++/MFC, ATL/COM
- Container Routing Support System, SeaLand, USA: Visual C++, ATL/COM, ActiveX controls, Windows NT Services, XML.
- Electronic Orders Input System, SeaLand, USA: Visual C++, Pro\* C/C++, Oracle Objects for OLE, ADO, Multithreading
- Property Appraisal software for US customer: Windows GUI Design, Visual Basic Coding Guidelines
- Patent and Annuities software for UK customer: SQLWindows, Windows API, Windows GUI Design
- Stores Management System for Indonesian customer: Visual Basic, Windows GUI Design
- VAX/Rdb COBOL to Unix/Informix COBOL Conversion software for Busy GmbH., Germany: C, ESQ/COBOL, Informix, Unix, VAX/Rdb

- Document Imaging Indexing System for ImageX, USA: Windows, Sun Solaris, Imaging software, Oracle, Progress
- Business Application projects (Register & Transfer of Shares, Personnel Payroll etc.) for Datamatics, India: Wang VS OS, COBOL, BASIC

#### Others (Description: Technical Areas Involved)

- Wrote a GUI Design guide for Business Applications for Mastek, India.
- Designed and conducted training courses for well-known companies in Mumbai, India: Windows programming, C/C++ programming
- Instructor for in-house COBOL training for Datamatics, India: Wang-VS, COBOL

<https://raviiyerteaches.wordpress.com/> (home page) contents given below:

#### Ravi S. Iyer's Software Lab. Courses

RAVI IYER TEACHES PROGRAMMING

Home

Last updated on 21st July 2016

This blog has been created to offer the content of software lab. courses taught by Ravi S. Iyer, Software Consultant, Puttaparthi, India, while he was offering free service as Honorary Staff/Honorary Faculty/Visiting Faculty to a Mathematics & Computer Science department in a deemed university in Andhra Pradesh, India, from 2003 to 2011. To know more about software background of Ravi S. Iyer, please visit: <http://eklavyasai.blogspot.in/p/about-me.html>.

The course contents in this blog/site typically include:

- Course structure
- Course book(s); Most courses are based on a primary course book
- Reference books and other resources, if any
- Teaching material (could be prepared by me and/or be a re-use of (reference to) external publicly available material usually from US university websites)
- Assignments

Given below are the course contents for regular courses:

1. C++ Programming, <https://raviiyerteaches.wordpress.com/2013/08/15/cpp-programming/> (short link: <http://bit.ly/cpp-prog> )
2. Advanced Unix Programming, <https://raviiyerteaches.wordpress.com/2014/03/18/advanced-unix-programming/> (short link: <http://bit.ly/au-prog> )
3. Unix Network (socket) Programming including pthread Programming, <https://raviiyerteaches.wordpress.com/2014/03/21/network-socket-programming-including-pthread-programming/> (short link: <http://bit.ly/net-prog> )
4. Minix Kernel Internals, <https://raviiyerteaches.wordpress.com/2014/03/30/minix-kernel-internals/> (short link: <http://bit.ly/minix-kern> )
5. Linux Kernel Customization – Mini Course, <https://raviiyerteaches.wordpress.com/2014/03/29/linux-kernel-customization/> (short link: <http://bit.ly/linux-kern> )

6. Java Web Programming (including HTML) – 2005 Course Report,  
<https://raviiyerteaches.wordpress.com/2014/04/10/java-web-programming-2005-course-report/>  
(short link: <http://bit.ly/java-web-p> )
7. Migration from C++ to C# – Mini Course,  
<https://raviiyerteaches.wordpress.com/2014/05/30/migration-from-c-to-c-mini-course/> (short link:  
<http://bit.ly/c-to-cpp> )
8. ASP.Net Web Programming in C# – Course Report,  
<https://raviiyerteaches.wordpress.com/2014/05/27/c-and-asp-net-web-programming-including-html/>  
(short link: <http://bit.ly/cs-asp-net> )

Given below are miscellaneous topics

1. Advice to Fresh CS Graduates & Post-Graduates on Industry Jobs; Prototype vs. production programming, <https://raviiyerteaches.wordpress.com/2014/08/12/student-project-prototype-programming-quality-vs-industryproduction-quality-programming/> (short link: <http://bit.ly/proto-vs-prod> )
2. Software development mini-project lab. courses – a report,  
<https://raviiyerteaches.wordpress.com/2014/08/12/software-development-mini-project-lab-courses/>  
(short link: <http://bit.ly/mini-proj> )

Please note that science and engineering/technology departments of some, if not most/all of, UGC (University Grants Commission), <https://www.ugc.ac.in/>, or AICTE (All India Council for Technical Education), <https://www.aicte-india.org/>, regulated educational institutions in India use a model of theory courses + lab. courses. In the deemed university that I was associated with, at Ist M.Tech.(CS) level it was usually 5 theory courses + 2 lab. courses in a semester, and somewhat similar at M.Sc. (Maths) level too. though perhaps it was only 1 lab. course per semester. In this deemed university, access to computer facilities for limited time was guaranteed for lab. courses teaching as well as student practice in the time table (the time table had 6 periods of almost an hour each from Mon. to Sat., if I recall correctly) with some additional time beyond the time table being allotted (evening hours). So, access to computer facilities was a significant constraint. During some of the later years of my stint, laptops were permitted for M.Tech. (CS) students and that reduced the constraints somewhat. [Please note that access to computer facilities in other UGC/AICTE institutions may be different from this deemed university that I was associated with. I am quite sure that it would be for significantly longer periods of time in the reputed UGC/AICTE institutions in the country.]

This blog is owned and operated by Ravi S. Iyer. This blog aims to share useful output of Ravi as a teacher of software lab. courses/programming courses.

### **Feedback from Former Students**

A former student who was taught the C++ Programming course by me in the deemed university in Andhra Pradesh, India, wrote the following over email on March 18th 2014 (modified slightly to fix a couple of minor grammatical errors):

*It is great that you have shared the C++ programming teaching material prepared by you on your blog. I really hope that people make use of it for I know how useful it could be. I can say this as a direct beneficiary of this, and today that is what fills my bank account at the end of every month.*

*I just hope and pray that Swami gives you the strength and determination to keep up the good work. I also think that I too can take a tiny part of the credit in this endeavour of you putting up these slides, as I was perhaps one of the many who would have suggested that you put these slides up in some forum accessible to the students' community. Thank you for considering the request.*

Another former student who had been taught Advanced Unix Programming and Unix Network Programming courses by me, wrote me on 22nd March 2014:  
Indian CS & IT Academic Reform (Past) Activism Blog Book

*These courses (Advanced Unix Programming and Unix Network Programming) went a long way in helping me land my job at Alcatel-Lucent. I had a one-on-one interview with my hiring manager that was entirely on Unix. After joining the company I learned that this person(manager) was a big time 'Unix fan'. It was very satisfying to have done well in that interview. On the job, we completely relied on Solaris Unix based servers and the concepts of processes and threads gained from these course(s), went a long way in helping me grasp the software.*

*Thank you Ravi Sir.*

### **Feedback from a Teacher**

An Assistant Professor of a deemed university wrote me in June 2014 that the slides and external links on this website/blog of mine is very useful. He further wrote that he is using them to teach BCA (Bachelor of Computer Applications) students.

---

## **Appendix B: Holistically Elite CS/IT Education vs. Technically Elite CS/IT Education**

Associated blog post date: 19th Jan. 2012, last updated on 21st Jan. 2012, link: <http://eklavyasai.blogspot.com/2012/01/holistically-elite-csit-education-vs.html> , short link: <http://bit.ly/holistically>

A holistic, service to society view would favour CS/IT education which combines technical know-how with application of this technical know-how to current socially relevant problems especially rural Indian society's problems as they are the most needy section of the country. Students who are taught in this fashion will have technical know-how and also experience the joy of serving the needy through CS/IT and earning their love. Such students may, later on in life, not only flourish commercially in industry or in academia but also contribute back to society, especially its needy, through software solving its problems, thereby leading to inclusive and harmonious growth of the community at large.

Some of the points touched upon in this post are general to higher education and not specific to CS/IT. But I am qualified by industry work & teaching experience and academic lab course teaching & projects' consultant experience only in CS/IT. So I make specific suggestions mainly about the CS/IT academic stream. I would also like to state that CS/IT being an applied field may have direct service to society application possibilities as well as research possibilities. Pure science fields like Physics, Chemistry, Bio-sciences or Mathematics may contribute via fundamental research. I do not have significant knowledge about these fields to comment on them. But by no means am I implying that their fundamental research contributions are "any lesser" in service to society than applied CS/IT software development contributions or CS/IT research contributions.

For CS/IT students to develop such a country-citizen/world-citizen, broad-minded outlook some time has to be devoted to teaching them application of human values/ethical values/spiritual values to CS/IT field. Some, perhaps most, educational institutions may prefer to be secular and not get involved with spirituality/religion but will certainly want to promote moral/human/ethical values. The financial system challenges that are threatening Europe, US and the world in general has its roots in grossly unethical & immoral actions of the financial system elite. The brunt of the suffering is not borne by the financial system elite who caused the problem in the first place but by the poor commoners who were taken for a royal ride by the financial system elite. So I think that all educational institutions, whether secular only or a

combination of secular+spiritual, will agree that for the betterment of the world, education in moral/human/ethical values is urgently needed, especially at higher education levels, as it is the lack of these values in the "highly educated" financial system elite which has brought the global financial system and the global community in general to such a precarious state.

But some educational institutions focus on pursuing excellence in specialized fields only and achieve fame in those specialized fields. It is these "elite" institutions that are revered by society today and almost every other educational institution tries to emulate them. I learned a new term the other day, "Ivy League Envy", from this Economist, Dec. 10th 2011, article, Schumpeter - University challenge: <http://www.economist.com/node/21541398>. The students from these "elite" institutions typically get the highest-paid fresher jobs in their area of specialization. Money earned by students and not their contribution to society, especially its poor, is the typical measure of success for most of the freshly passed out students and their parents. In India, for the CS/IT field, it is the IITs & now, probably, the IIITs which fall in this "elite" category.

In my view, these are "elite" from a speciality-excellence point of view only. There are other educational institutions which are "holistically elite" giving the "right", in my humble opinion :, balance of speciality-knowledge & human values knowledge.

I guess there will always be hallowed portals of "elite" education both at material/secular levels [MIT, Harvard, Oxford, IITs etc.] and at spiritual levels [Elite institutions among Christian seminaries, Muslim Madrasas, Jewish seminaries, Buddhist monasteries, Hindu monasteries etc.]

A few educational institutions go for a combination of spiritual/moral/ethical, cultural, sports & 'secular' education as they want to give both an education for life (truthful, ethical, peaceful and loving way of life - moral/ethical/spiritual life) & an education for a living (material life). These holistic educational institutions cannot aim for the same level of excellence in secular education like that of the elite secular only educational institutions. But it must be noted that they achieve excellence by holistic standards and so become a hot destination for parents & students desiring holistic education. So they are "holistically elite" as against being "secular only elite".

But some parents and students may want to avoid any value based education, especially at higher education levels like CS/IT graduate/post-graduate level. There should be NO compulsion. Such parents & students can steer clear of value-based educational institutions. Anyway, as of today, value-based higher education institutions are a very small minority.

Another vital factor is the "big paycheque" motivation for students & parents. The "holistically elite" CS/IT fresher graduate would get a decent paycheque but it would typically be significantly lesser than the paycheque of the "technically elite" CS/IT fresher graduate. So students & parents interested in a "big paycheque", also can steer clear of value-based educational institutions.

Then we have an issue of best education for intellectually-elite students versus best education for intellectually-commoner students. [Note that an intellectually-elite student may be weak in social/humanistic skills, whereas an intellectually-commoner student may be very talented in social/humanistic skills i.e. he may be a social-humanistic-nature-wise-elite student.] Some intellectually-talented educators would like to focus on creating the best education system for the intellectually-elite.

In my humble opinion, we should not suffocate the elite and force them to learn in commoner student ways. Society must allow the crème de la crème to pursue excellence in its chosen fields. We must keep alive, within bounds of reason and social acceptability/relevance, the portals, physical and digital, which inspire, ignite and elevate minds to achieve excellence. Such excellence can be in secular fields like Physics, Evolution, Computing, Medicine, Sociology, Economics and also in spiritual fields. Why limit ourselves to only Darwin, Newton, Neils Bohr, Einstein ... We should also have beacons of love, joy, peace & light like

Jesus, Buddha, Mohammed, a Mahatma Gandhi, a Mother Teresa, a Pope John Paul II, a Ramana Maharishi ([http://en.wikipedia.org/wiki/Ramana\\_Maharshi](http://en.wikipedia.org/wiki/Ramana_Maharshi)) etc.

But we should not ignore the non-elite, the 99 %, as the Occupy movement refers to itself. We need solutions to cater to their educational needs as well. That is ABSOLUTELY VITAL for social peace and harmony. The intellectually-elite can have the freedom to pursue their area of excellence only if social peace and harmony is ensured.

In conclusion, I feel that "holistically elite" CS/IT education is what society needs today far more than "technically elite" CS/IT education. The amount of money that students earn after they finish their education should not be the primary measure of success of the "holistically elite" educational system. The contribution made by students using the CS/IT skills they have gained from academia, after they finish their education, or in some rare cases, even while they are undergoing their education, to society's needs, especially the needs of society's poor, and how much love, joy & peace they bring to their individual life, their family and the community in general should be the primary measure of success of the "holistically elite" CS/IT educational system.

---

## **Appendix C: Discussion and Debate on lack of Proper Warranty for Software, and need thereof**

### **The Without Warranty Wild West Software Industry**

Associated blog post date: 18th Mar. 2013, last updated on 28th Mar. 2013, link: <http://eklavyasai.blogspot.com/2013/03/the-without-warranty-wild-west-software.html> , short link: <http://bit.ly/the-without>

The world today is pervaded by a huge variety of software from Operating System software to embedded devices software to mobile phone apps. Serious bugs including very scary security flaws in these software are commonplace and accepted as an inescapable reality and risk of using software. Is that a proper stand for the software industry and software customers & users to take?

IMHO, the software industry culture is to blame for the current rather unsure and unprofessional state of the software world. In the nearly three decades that I have seen it, been part of it and experienced it as a user of software, the software industry worldwide seems to operate as a Without Warranty Wild West industry, with current generation of released software seeming to have far more bugs/problems than software on Mainframes/Mini-computers two to three decades ago. Forget application software bugs, the very operating system on which the whole software stack runs is without warranty and has a host of bugs including some terrifying security weaknesses. If you get hit by a malicious virus then it is your bad luck, that's it. You cannot hold the software company accountable. It is this lack of accountability of the software industry that, IMHO, is at the root of the excessive software failures that users have to suffer from.

Once the software industry becomes like other mature engineering industries who are accountable for their products (e.g. a Television set manufacturer is accountable for its product), then, IMHO, the entire software ecosystem including the teaching/academics part of it will be forced to become more "professional" about software development/engineering. But will most software companies, or perhaps any software company, offer warranty for their software easily and of their own accord? I strongly doubt that will happen without outside or public pressure/intervention.

Meanwhile we have to continue to live with the Without Warranty Wild West software world and produce software applications, within time & money constraints, of reasonable quality with some bugs here and there being tolerated like some security weaknesses here and there in the OS itself are tolerated. Is this the ideal software world? Certainly not! But can we freeze application software development till the software world becomes ideal? I don't think so! As far as I know, very, very few people like Prof. David Parnas, <http://www.amadon.ca/Public/information.htm>, and Prof. Bjarne Stroustrup, <http://www.stroustrup.com/>, are thinking, writing and talking about these fundamental issues with software development today. And, IMHO, most people in the software industry or software academy don't seem to bother about or, in perhaps the vast majority of cases, even know about, what they are saying!

The result is that, in this far-from-perfect software world, at least for a wide range of application software, as a software developer one also has to join the crowd and try to deliver software quickly, using approaches like component based development, even if the produced software has a few hidden bugs here and there. One may take a lot of care to ensure that one's code is well designed and reasonably tested given the time constraints one is operating under, but one simply cannot control the bugs in the components that one uses for the solution. As an individual, you can't stop or change this "some tolerance for bugs in return for quick and cheap software solution" culture. You, normally, have to go with it. Or else you, as an exception, can simply stay away from providing such solutions. Others then will satisfy the market demand for such solutions by providing them in your place. I could be wrong but that's the way I think it is.

---

I have also added below snippets of my part of a conversation that I had with a software development practitioner on this topic.

I continue to hold the view that, at least for paid/significant cost software, providing a warranty like other fields of manufacturing, will bring order to the chaos that pervades the software industry today. Whether it can be pushed through or is practical enough are different matters. But if you ask the hapless user s/he does not care about whether it is practical enough or not, s/he is fed up of paying significant amount of money for software and having to live with painful bugs in what s/he has paid for!

...

About poor workmen (i.e. poor quality software developers/software engineers) being a problem, I agree. But till the poor quality of output produced by poor workmen is penalized, the market/industry will employ such workmen and make money at the expense of hapless software customers & users. Prof. Parnas and others seem to hold the view that introducing licensure for software engineers will control the poor workmen problem like in other engineering sectors in the Western world. Maybe it will. But despite tremendous efforts of Prof. Parnas and others they, it seems, have not succeeded so far in the Western world adopting licensure for software engineering/development in a big way.

...

I don't want to get into a detailed discussion on the practicality of warranty for software as I have not examined the matter in depth (and neither have the time to do so now). But then let us at least acknowledge that the software industry folks (including me in the past and perhaps in the future as an Open Source developer) are amateurs as compared to professionals from fields like civil engineering or mechanical engineering. Yes, most of us are/were highly paid for our software development work but we are/were only highly paid amateurs. We don't have the right to call ourselves "professionals" as we cannot give any warranty/guarantee about the software solution we provide, in most, if not all, cases.

Please note that warranty does not mean error-free. A Television set may develop faults within its warranty period. Warranty, in my understanding, implies accountability to the extent of replacing a faulty product with a working product at no extra charge to the customer and also compensating a customer for significant damages incurred due to a faulty product.



A distinguished US academic had a small mail exchange with me on this topic. I would like to share my part of that conversation below as I feel it adds some value to the topic.

The point raised was that we all know today's software is poorly engineered. How do I propose to solve the problem?

My response was:

I think software companies need to become accountable by, for example, accepting financial penalties for significant failure of software during a warranty period. Catastrophic failure of software repeatedly should involve a govt. approved industry body examining the software very much like medical bodies examine repeated catastrophic failures of medical doctors. If the industry body deems that the software deployed did not follow minimum practices for design, code, test, etc. then a financial penalty should be levied on the company, and the record of the company and concerned software developers should reflect this failure. This will create a fear of appropriate professional repercussions for poorly engineered software.

As of now, I don't think there is this kind of professional penalty/blot-on-record fear either at the company level or at the individual software developer level. The focus is on getting the minimum stuff done for the customer to sign on the dotted line and make the payment for the customized software developed/product sold. In case of products, the fear is of bad publicity if the product is bad, and so losing out to the competition - that may be a motivator for trying to ensure good quality. In case of customized software solutions, the fear really is of repeat business not coming. IMHO, such fear is nowhere close to the fear that a medical professional has if s/he is involved in negligence or malpractice. We need to bring in this fear of repercussions like in medical practice for badly done software. That may really bring some professional like discipline in software engineering/development.

I hope I am not sounding like some fear-mongering dictator :). IMHO, very unfortunately, without fear of significant repercussions, people, in general, tend to cut corners in almost all walks and activities of life. In Asian countries like India, it is very much the case but I think it applies to quite some extent even in the materially advanced Western countries.

Here's a mail exchange (slightly edited) with another correspondent/friend who kindly permitted me to share it on this blog post.

Friend wrote in response to this blog post: I have one quick observation : When comparing a TV set with a S/w, the manufacturer of a TV will replace the set within warranty period if the operating conditions have not changed.

My (Ravi) response: Well, I guess the warranty will be valid even if operating conditions change but are within the operating conditions for which the warranty is valid.

Friend: Let us say that a TV worked for 3 months and suddenly kicked its bucket. Then he will replace it - however, it should not be because of usage during high / low voltage, pouring water into it, etc.

Ravi: Yes, IMHO, as then the conditions for the warranty would be violated.

Friend: In the case of S/w, is not the scenario different?

Ravi: There certainly will be certain differences between software and an electronics gadget like a TV. But I was going mainly by the principle which I feel can be applied to software too.

Friend: If you have been using the same functionality/features and if the S/w had worked earlier can it really stop working?

Ravi: Yes. The s/w may encounter new data which breaks it (e.g. boundary conditions). Further, nowadays for some software like operating system and anti-virus software you have the automatic updates feature, say over Internet, which may introduce new buggy code into the software, even a few days or weeks after it is installed.

Friend: One scenario is as follows:

For instance, you might use a new choice in a menu and find that it is not doing what it is supposed to do, but perhaps it was not working right from the beginning - just that we did not find out about it.

Ravi: Yes, this is clearly a possibility with today's software which are sometimes overloaded with features/functionality that one does not regularly use.

Friend: For instance, in a TV which claims to have 1000 channels, perhaps we tuned only the first 50 ones and after 6 months when we try to tune the 51st channel we find it is not possible. The above case is similar to this. So if we find it out within the warranty period TV manufacturer replaces it and so should the S/w provider / vendor.

Ravi: Agreed.

Friend: Another scenario is what I wrote first. In a s/w can it happen? If it does, is it problem with H/w or S/w or some other "element" of the system? Whose responsibility should this be?

Ravi: If it is s/w which is the cause e.g. not handling the new data, then the responsibility should be the software vendor's.

Friend: (I may not have) explained myself clearly, but I hope you see there is a difference here? Or is there??!!

Ravi: I am sure there will be differences. S/w warranty will be more complex than a TV warranty.

In this connection, ideally a vendor should publish its QA reports for the software it sells to users. The QA report must detail all the tests that the software was put to. Such clear documentation of the QA process will allow customers to get a feel of the rigor of the software vendor's QA process. If the customer discovers a bug later on s/he can check against the QA report and see how this bug slipped through. The customer will then know whether the QA report missed it or whether it stated that it passed that test. In the latter case, a question would pop up about the truthfulness of the QA report of the vendor.

Such transparent QA reporting will give the customer a feel of how robust the software is, thereby allowing him/her to decide how much to depend on it.

I don't know whether such QA reports are publicly made available for software products now. Maybe open source guys do that. Not sure if a company like --- would do that though :). They will consider it perhaps only if government (i.e. legislators) or courts force them to do it.

... Another mail exchange with same friend ...

Friend: Actually, what I find very very interesting is that so far (a software industry body) or any such body has not thought of this.

Ravi: Perhaps they know about it. (They may not want to consider) any demands for warranty as they may feel that it would negatively impact software companies, which they represent.

Friend: Further, the non-IT MNCs which pay such huge sums also do not seem to be reporting the failure of s/w as a problem - surely it drains them of millions.

If one particular part is faulty, a car manufacturer takes the other party to task, but somehow when we come to s/w everybody seems to have infinite tolerance!!

Ravi: I think the world at large has been dazzled by the achievements of software. That and the huge money power that software companies have, makes these companies very formidable entities to aggressively question. I don't think governments will have the (will) to push software companies hard anywhere in the world today as the world is becoming more and more dependent on software, and (software companies would be having, I guess, significant political lobbying power). It is the Western world justice system that has the (will) to question and even fine software companies - e.g. ---,--- being fined for their practices like restrictive trade practices (---) and for using/capturing data of people from their wireless (home) networks without their permission (---).

I think it is just a matter of time before some software glitch results in a catastrophe, provoking somebody/some entity to take the matter up to a Western world court (I don't know whether it has happened before; maybe it has but as I was not following this space before the past year or two, I do not know of it) - the top Western world judges will not easily buy arguments of software companies that the software is without warranty/"At your own risk" and so they are not legally liable for anything related to their software. They may (apply) some other law.

Friend: Thanks for this really thought-provoking perspective.

Ravi: You are welcome, brother --.

-----

Given below is a comment from blog post associated with above article.

Ravi S. Iyer wrote on March 22, 2013 at 1:25 PM

Even NASA is not able to avoid software glitches. Here's info. about glitch(es) that has/have stalled its latest \$2.5 billion (reportedly) Mars Rover Curiosity probe/robot/mission: Time article (archived link: <https://web.archive.org/web/20130323015409/http://science.time.com/2013/03/19/mars-rover-curiosity-stands-down-after-new-problem/>), short link: <http://bit.ly/3c3bcGr> ) and Information Week article (archived link:

<https://web.archive.org/web/20130321094012/http://www.informationweek.com/government/information-management/nasa-curiosity-rover-hit-by-software-sna/240151092>, short link: <http://bit.ly/32p2RrW> ).

### **A Debate on Warranty for Software**

Associated blog post date: 24th Mar. 2013, last updated on 28th Mar. 2013, link:

<http://eklavyasai.blogspot.com/2013/03/a-debate-on-warranty-for-software.html> , short link: <http://bit.ly/a-debate>

The post, The Without Warranty Wild West Software Industry (previous article in this book), whose contents I mailed out to some correspondents, resulted in a few of them responding with the problems that

warranty for software has, and some related views of theirs. I have given some of the responses below, and then my response to them.

Correspondent1 (Corr1) is 'someone with decades of experience in academia and industry'. He wrote:

This is an area I have worked in for decades so it is of great interest.

I think the basic problem you are ignoring is that most software is developed to meet imprecise and incomplete requirements and specifications. I can give you numerous cases where the specifications failed to define all the possible cases that some software needs to cover and left it open to the implementer to decide what to do. There are other cases where assumptions are made but not stated and yet other cases where the specifier just did not know the existence of the case and/or did not know what to do if it occurred.

We are talking about immensely complex systems, far more complex than any machine that is built. Just look at the billions or trillions of states that a large complex system will go through in its lifecycle.

I think that given the limitations of the field today, it is not correct to talk about warranties because a warranty specifies a performance against a specification. However, we are far from having anything like a precise and complete specification for any software system.

Instead, the focus should be on the process used to produce the software: the kind of design, the steps used for its construction and the tests done during the development. All this needs to be recorded and saved for ever.

In the 1980s in the UK, the defence authorities were persuaded to issue standards called DEF 0055 and 0056 for software. The standard had a lot of conditions that no developer could hope to meet (including one that required all tools used during development to themselves have formal proof of their capabilities). The standards have since been made more realistic.

You may be interested to read what Nancy Leveson has written on software and systems reliability (she has written a whole book on the subject), likewise Peter Neumann and others. Their main objection is to the lack of rigorous method during software development, the sort of due diligence that is essential and often watered down or ignored.

Another correspondent (Corr2) wrote that the problem starts at specifications level as software is very poorly specified. He says that Formal Methods can make specifications precise but that industry does not accept it. To see the exact words of his view, please see blog post associated with this article.

My (Ravi's) response to the above:

Yes, I agree with both Corr1 and (Corr2) that it is a very difficult, if not impossible task to define precise specifications for most of the complex software that is developed and used nowadays.

But I still feel a limited warranty of some sort could be considered. Now, I am not an expert on this matter and have not studied the work of researchers on this topic. However, IMHO, I am entitled to express my opinion based on my nearly three decades of experience of the software industry and software academia.

The test cases, test reports and test data that are used by a software vendor to check the quality of the software prior to releasing it to the customer, should be handed over to the customer. This will enable the customer to independently, if the customer so desires, check the quality of the software for themselves. Further, such test information & data can constitute the operating conditions for

which the software vendor can provide a warranty. These operating conditions could also specify the version and other details of software components supplied by other vendors that the software solution is based on. E.g. Database, Report Generation tools and the Operating System.

If the customer encounters a bug for a specific use case in the delivered software system, they can check whether the test cases and other operating conditions cover such a use case. If it does cover it and the use case does not work as required in the test environment using the specified/agreed-upon test data, even though the test report claims that it worked, then the customer has a clear case of the software vendor not having tested the software properly as per the agreed software development contract (or software product contract with the variation that the software product company unilaterally fixes the test cases, test data, environment etc.) The limited warranty will come into play with the software vendor having to do whatever is specified in such a warranty, e.g. fixing the bug(s) at no charge to customer, and, in the case of a very-confident/aggressive warranty, provide compensation for loss faced by the customer due to the bug(s).

But will such a limited warranty which applies only to test cases & test data etc. considered at software development time be of any use at all? I think it will. As of now, I don't think that, in typical software deliveries, such test details are given to customers in such a way that they can run the tests independently. Once such a culture gets introduced, customers will get more involved with the testing process. Specifically they may ensure that the test data covers all the cases that they want in the software, as they get a warranty for such test data. If for their test data the software is found to not work during period of warranty they can hold the software vendor accountable! That can give tremendous confidence to customers about some sort of limited accountability for the software, as compared to the situation now where essentially the customer is told that the entire software is to be used "at your own risk"!

Just imagine the business risks that come into play when a business becomes heavily reliant on such software which is used "at your own risk". Further try to imagine a chairperson or managing director of such a business trying to explain to angry shareholders in an AGM, a catastrophic software failure that caused significant damage to the company, and steps that will be taken to avoid similar catastrophic software failures in future.

Now software vendors offering such limited type of warranties may charge extra for the warranty. That is fair, IMHO. The customer gets a choice between less expensive software but without any form of warranty and somewhat more expensive software with some sort of limited warranty. This would parallel many other fields in India. E.g. mobile phones. The customer can buy cheap mobile phones produced by almost unknown company(ies) with no warranty OR mobile phones produced by reputed company(ies) with warranty which are significantly more expensive than the former. There is room in the mobile phone market for both.

Similarly, I think, if some type of limited warranty is offered for some software, over time, we will have both types of software solutions - no warranty and limited warranty, each with their own space in the market.

I should also mention that many devices/machines which use embedded software (e.g. medical devices, some or most cars, airplanes), I presume, are being sold with a warranty for the devices/machines which will include the embedded software part. But that is a specialized use of software, and anyway they seem to be providing a warranty already. It is the only software solution vendors (no device) that I am referring to in this post (and previous post mentioned at the top of this post). Now, in the only software solution space there is a huge variety of software from Operating Systems to Compilers to Database-oriented business applications. While I was writing the above my thoughts have focused more on Database-oriented business applications, as life in Indian towns & cities today seems to be heavily interacting with or dependent on such applications (e.g. banking,

mobile top up, railway reservation, hospital admission etc.) but I have tried to be generic. So a lot of it may apply to other kind of software like a mobile spreadsheet application too.

Now about importance to be given to the software process (requirement analysis, requirements capture, design, code, test, deployment etc.) - I entirely agree. Ideally there should be a minimum standard or a set of few minimum standards for the software process which should be published by government approved industry standard bodies. When a software vendor takes up a software development order it should inform the customer about which industry standard process it is following. Further, key artifacts produced as the software is developed using the industry standard process (design specifications, program specifications etc.) should be provided to the customer, so that the customer can, if needed, itself, or by using a third party, inspect the artifacts to check whether the specific industry standard process is indeed being followed.

Like in the limited warranty suggestion, the customer can be given a choice: less expensive software development but which may not follow any industry standard process OR more expensive software development which follows a specific industry standard process and with various artifacts produced by the process being provided to the customer.

An important and significantly large exception to the above is that some software solutions are proprietary to the vendor and so most artifacts produced by any software development process followed for a software solution will have proprietary information that the vendor will not want to share with the customer.

Regarding Formal Methods and (other) techniques to produce bug-free software, my knowledge about these topics is very, very limited. I believe such methods and techniques have not yet been widely adopted by the large majority of software solution providers. Perhaps there are some significant challenges/drawbacks which are impeding their widespread adoption or perhaps there are other political/mind-set type reasons for it. Anyway, their adoption is something that I am not in a position to contribute to in any way. I go by what the mainstream software development industry follows. If and when they switch to Formal Methods in a big way, I may have no option but to consider it seriously then.

A correspondent brought up the "fit for its intended use/purpose" Engineering concept.

Ravi: I did some reading up about it. [http://en.wikipedia.org/wiki/Implied\\_warranty](http://en.wikipedia.org/wiki/Implied_warranty) states (for USA):

In common law jurisdictions, an implied warranty is a contract law term for certain assurances that are presumed to be made in the sale of products or real property, due to the circumstances of the sale. These assurances are characterized as warranties irrespective of whether the seller has expressly promised them orally or in writing. They include an implied warranty of fitness for a particular purpose, an implied warranty of merchantability for products, implied warranty of workmanlike quality for services, and an implied warranty of habitability for a home.

...

An implied warranty of fitness for a particular purpose is a warranty implied by law that if a seller knows or has reason to know of a particular purpose for which some item is being purchased by the buyer, the seller is guaranteeing that the item is fit for that particular purpose.[1][2][3]

...

In some jurisdictions, an implied warranty in a sales contract can be expressly disclaimed by the use of specific language, such as the words, "as is" or "with all faults".

[Wiki References:]

1. Primack MA. (2009). Representations, Warranties and Covenants: Back to the Basics in Contracts. National Law Review.
2. "Difference between a Guarantee and Warranty". Archived from the original on December 5, 2016. Retrieved December 5, 2016.
3. Ferrara LN, Philips J, Runnicles J. (2007). Some Differences in Law and Practice Between U.K. and U.S. Stock Purchase Agreements. Jones Day Publications.

--- end wiki page extracts ---

So warranties need not always be limited to performance against a specification. A flawed car tyre that leads to many car accidents may not have failed its specifications, but would have failed the "fit for its intended use/purpose" implied warranty and so the manufacturer would be held accountable by the legal system/government.

The software industry seems to have disclaimed the implied warranty of "fit for its intended purpose" by usually adding the words "as is" in the contract.

Okay! So the software industry is legally not accountable even for any implied warranty. But how long can this "unaccountable" thing go on? I think it is just a matter of time before some government or some judiciary somewhere in the world responds to one or more catastrophic software failures badly affecting large number of people, by imposing accountability on software solution providers. And such imposed accountability, in all probability, ain't gonna be pretty!

It would be far, far better if the software industry puts its own house in order by experimenting with some sort of limited accountability. My suggestions outlined earlier may or may not be good enough. There may be better suggestions somewhere else. Or, if nothing good enough is around yet, if the software industry puts its mind to the problem, I am sure they can come up with some concrete suggestions for limited accountability (with its possibly higher costs which the customer must be willing to bear) and which they can experiment with.

Another correspondent questioned my assumptions about a viable market for the more expensive limited-warranty software.

My (Ravi's) views on it are as follows:

The impression that I have is that people think that software can never have any sort of even limited warranty. As far as I know, concerted efforts by software industry to show that some sort of limited warranty can even be considered, are absent.

I think for software that companies and government become heavily dependent on, e.g. hospital management software which controls many aspects of hospital functioning, income tax return online filing etc., both companies and government may be very interested in more expensive limited-warranty software solutions. In fact, they may consider only such solutions, if they were available. I am not saying that vendors should be forced to offer limited warranty software - customers should be able to specify that they are only interested in limited warranty software solutions. As of now, it seems to me, that no software (only) solutions or insignificantly small number of software (only) solutions fall in this category and so customers will not be able to specify in their Request for

Proposals/(Request for) Tenders for software solutions that they want limited warranty software solutions only.

Another point was about whether the service contract offered by software solution providers does not meet the need.

My (Ravi) views:

My initial thoughts are that typically a service contract comes into play after the warranty period runs out. Further the service contract is more limited than the warranty - e.g. may not involve replacement of product at no charge.

To a TV manufacturer kind of guy, the software industry support contract may seem like milking the customer for the flaws in your product right from day one of the sale! I think a mature product/solution offering should have a period of free repairs/replacement, and I think warranty seems to be the appropriate name for it.

A larger issue is what if the product/solution is discovered to be not "fit for its intended use/purpose" some period after the sale is made/software is used. A warranty seems to imply far more effort and cost to handle such situations than a service contract.

A correspondent wrote that disclaiming the implied warranty of "fit for its intended purpose" (or to be more precise, "fitness for a particular purpose", see another definition also using the same phrase as the wiki page: <http://www.businessdictionary.com/definition/implied-warranty-of-fitness-for-a-particular-purpose.html>) with an "as is" clause may not really work/be valid (in all countries/jurisdictions). He mentioned that he had to learn about such matters when he took the exam for a Professional Engineering license (in Canada, I believe). The correspondent wrote that if somebody presses the issue then the implied warranty liability will hold even if there is an "as is" clause (at least in some countries/jurisdictions like Canada).

Correspondent Corr1 wrote:

"warranty of fitness for a particular purpose"

The whole point about large and complex software systems is precisely that the "particular purpose" is (a) not defined, (b) incompletely defined or (c) wrongly defined.

Moreover, what works with one version of infrastructural software (operating system, database system, communication system etc.) may not work with later versions.

Of course we need more rigour in the way software is defined and produced. This is something software engineers aspire to and it will take time. Nevertheless, good software engineering practice today already follows best practices and uses all available tools to make software reliable.

-- end Corr1 message --

Ravi's response: The particular purpose of a car tyre too may not be defined completely. But that does not allow the tyre manufacturers to escape accountability for repeated accidents involving their tyres. We have a famous case of millions of tyres manufactured by a particular company being recalled, [http://en.wikipedia.org/wiki/Firestone\\_and\\_Ford\\_tire\\_controversy](http://en.wikipedia.org/wiki/Firestone_and_Ford_tire_controversy), [http://www.thecarconnection.com/news/1061100\\_a-decade-after-the-firestone-recall-tire-safety-still-newsworthy](http://www.thecarconnection.com/news/1061100_a-decade-after-the-firestone-recall-tire-safety-still-newsworthy). The reports do not mention that the tyres did not meet the specifications. They apparently did but, perhaps for cost saving purposes, lacked a crucial extra liner on the tire which would have made it safe and car/road worthy.



I feel that the software industry needs to seriously explore some sort of limited accountability instead of sheltering in an "as is" or "use at your own risk" completely unaccountable strategy because the specifications for software typically are not well defined. The limited accountability can take into account the fact that specifications for software are typically not well defined and the version changes of infrastructure software (OS, Database, Comm. s/w etc.) E.g. The accountability could be limited to only the specific versions of the infrastructure software that was used for development and test of the software solution, and could be limited to the functionality specified in the requirement specifications and test specifications/reports.

Given below are comments (slightly edited) from the blog post associated with this article.

***Ravi S. Iyer wrote on March 28, 2013 at 3:20 PM:***

I came across a very interesting quote attributed to (don't know if the attribution is accurate) Abraham Lincoln related to dangers of power craze/power abuse. I think this applies to the top people in powerful software companies, and the powerful software developer/engineer community in general too. While they may not have explicit political power, IMHO, they have extraordinary implicit power due to the vast and pervasive spread of software into virtually all aspects of life today. The political and other powers may turn to the software community for help and support, and so may be very reluctant to challenge them.

“Nearly all men can stand adversity, but if you want to test a man's character, give him power.” - Abraham Lincoln, [https://www.brainyquote.com/quotes/abraham\\_lincoln\\_101343](https://www.brainyquote.com/quotes/abraham_lincoln_101343).

Perhaps we now have a test of character for the worldwide software development/engineering community/industry. Can we rise up to the challenge and become a profession of "character"?

***Ravi S. Iyer wrote on March 28, 2013 at 9:16 PM:***

I felt it appropriate to share a personal experience regarding possibly poor car tyre quality. Please feel free to skip it if you are not interested. I was driving a Japanese model car, IFIRC, in the late 80's on a highway in New Hampshire/Massachusetts, USA, commuting from residence in Nashua, NH, [http://en.wikipedia.org/wiki/Nashua,\\_New\\_Hampshire](http://en.wikipedia.org/wiki/Nashua,_New_Hampshire), to business/work at Wang Labs., Lowell, MA, [http://en.wikipedia.org/wiki/Wang\\_Laboratories](http://en.wikipedia.org/wiki/Wang_Laboratories), at quite some speed, when one of its tyres gave way. I lost control of the car - it moved from the fast to the slow lane and then the breakdown lane. It was all set to hit a railing beyond the breakdown lane at still significant speed, when I tried the brakes. That resulted, fortunately, in the car turning back from breakdown lane to slow lane then fast lane, other (breakdown) lane, and then gently down the median valley. The car then turned upside down in the median valley but at gentle speed and came to a halt. Miraculously neither I nor two other passengers in the car including one in the rear seat who did not have a seatbelt on, had any grievous physical injuries. We were also very lucky that the rush hour traffic immediately behind our car had noticed the problem and stopped their cars and the traffic behind them on all lanes.

Of course, I was given a dressing down by the passengers of the car, and since then I don't have the same level of confidence that I had earlier about cars.

Now I feel maybe I had a defective tyre on the car. The car was from a rental agency. They gave me/us another car and that was it. As I was young then and a foreigner to the USA I did not press anybody to know the proper cause of that incident but just thanked my stars that I and others had come out unscathed from the incident/accident. I mentioned this experience to convey how the value of engineering quality, tyre

engineering in this case, was brought home to me personally in a way that I will not forget for the rest of my life. And in this context, the "fit for its intended use" phrase/clause conveys the engineering quality aspect so well. Perhaps the tyre was not "fit for its intended use". But the cause could have been something else too. I feel I did not drive recklessly and so was not at fault but then that is just my opinion.

*Ravi S. Iyer wrote on April 1, 2013 at 10:28 AM:*

This article, dated Aug. 2004, over eight years ago, shows how managers like a GM CTO viewed the matter then. The article also has views of software company people who have a counter view. Enterprise Software: Warranty Woes, <http://www.baselinemag.com/c/a/Business-Intelligence/Enterprise-Software-Warranty-Woes/> (short link: <http://bit.ly/2v8lC7b> ).

The essence of the article is that the CTOs wanted to have a proper warranty but the software companies were not interested in providing a proper warranty. In its concluding portion (on page 5), the article states, "Large companies will have to shun software vendors that won't assume any liability for malfunctioning software.

Small- and medium-sized companies will have to band together and insist on a standard set of warranties from providers of off-the-shelf software that at least guarantee the buyer that applications will be free of code defects and be secure outside a lab."

Well, eight years plus have passed since the article was published and we seem to be having roughly the same state of affairs.

## **Appendix D: List of some other eklavyasai blog posts including on “How to Teach Programming?”**

### **Improving the Practice of Software Development in Indian CS & IT Academia**

1. Suggestions to Improve CS & IT Education in India sent to Think India - A Public Policy Advocacy Think-Tank, <http://eklavyasai.blogspot.com/2013/12/suggestions-to-improve-cs-it-education.html> (short link: <http://bit.ly/suggesti> ), Dec. 2013.
2. NBA India: World Summit on Accreditation - Themes include Bridging Academia & Industry and Outcome Assessment Tools, <http://eklavyasai.blogspot.com/2014/03/nba-india-world-summit-on-accreditation.html> (short link: <http://bit.ly/nba-india> ), Mar. 2014.
3. Eminent Indian computer scientist-cum-administrator Dr. S. Ramani writes: this Indian CS & IT academic reform activism blog is interesting, <http://eklavyasai.blogspot.com/2014/07/eminant-indian-computer-scientist-cum.html> (short link: <http://bit.ly/eminant-i> ), July 2014.
4. Dangers of pro-MATLAB-research and anti-software-development Computer Science higher education policy, <https://eklavyasai.blogspot.com/2015/02/dangers-of-pro-matlab-research-and-anti.html> (short link: <http://bit.ly/dangers-of-p> ), Feb. 2015

### **How to Teach Programming?**

5. Programming: Practical Solution Mindset vs. Scientific Mindset, <http://eklavyasai.blogspot.com/2011/10/programming-engineering-mindset-vs.html> (short link: <http://bit.ly/prog-engg> ), January 2013.

6. Does a Teacher of Programming Need to Know Turing Machine?, <http://eklavyasai.blogspot.com/2011/10/does-teacher-of-programming-need-to.html> (short link: <http://bit.ly/does-teach> ), October 2011.
7. Comments on Stroustrup's paper, "Software Development for Infrastructure" in IEEE Computer, <http://eklavyasai.blogspot.com/2012/03/comments-on-stroustrups-paper-software.html> (short link: <http://bit.ly/comments-on> ), Jan. 2012; post last updated in April 2012.
8. Texas A&M University's Approach to Teaching Programming in (US) CS Academia, <http://eklavyasai.blogspot.com/2012/01/texas-universitys-approach-to-teaching.html> (short link: <http://bit.ly/texas-univ> ), January 2012.
9. CS & IT Academia: How to Teach Programming?, <http://eklavyasai.blogspot.com/2011/09/cs-it-chittar-comment-response.html> (short link: <http://bit.ly/how-to-t> ), September 2011.
10. CS & IT Academia: Some Comments on Teaching Programming, <http://eklavyasai.blogspot.com/2011/10/cs-it-academia-some-comments-on.html> (short link: <http://bit.ly/some-comm> ), October 2011.
11. US CS & IT Academia: Usually TAs/RAs Teach Programming, <http://eklavyasai.blogspot.com/2011/11/us-cs-it-academia-usually-tasras-teach.html> (short link: <http://bit.ly/usually-ta> ), November 2011.
12. CS & IT Academia: Should 'Customize' Instead of 'Build' be Taught?, <http://eklavyasai.blogspot.com/2011/11/cs-it-academia-should-customize-instead.html> (short link: <http://bit.ly/should-c> ), December 2011.
13. My C++ online test invigilator & evaluator experience of student blaming me incorrectly for his poor performance at vital entrance exam, <https://eklavyasai.blogspot.com/2017/04/my-c-online-test-invigilator-evaluator.html> (short link: <http://bit.ly/my-c-o> ), April 2017.

## CS & IT Research

14. Prof. David Parnas' views on Corruption of (Academic) Computer Science Researchers/Scientists by Publication Counts, <http://eklavyasai.blogspot.com/2013/04/prof-david-parnas-views-on-corruption.html> (short link: <http://bit.ly/prof-david-p> ), Apr. 2013.
15. Oct. 2013 Economist article on Problems with scientific research - How science goes wrong, <http://eklavyasai.blogspot.com/2014/01/oct-2013-economist-article-on-problems.html> (short link: <http://bit.ly/oct-2013-e> ), Jan. 2014.
16. Pursuing CS/IT Research Individually - Ph.D. Possibility?, <http://eklavyasai.blogspot.com/2011/09/pursuing-research-individually-phd.html> (short link: <http://bit.ly/pursuing-r> ), Sept. 2011.
17. CS & IT: Pure vs. Applied Research, <http://eklavyasai.blogspot.com/2011/09/cs-it-long-comment-8-1.html> (short link: <http://bit.ly/pure-vs-appl> ), Sept. 2011.
18. Google's Hybrid Research + Development Model, <http://eklavyasai.blogspot.com/2012/07/googles-hybrid-research-development.html> (short link: <http://bit.ly/googles-h> ), July 2012.
19. [Veteran Computer Science researcher's] Ground Rules for Sending Scientific Papers for Publication, <http://eklavyasai.blogspot.com/2012/12/ground-rules-for-sending-scientific.html> (short link: <http://bit.ly/ground-r> ), Dec. 2012.
20. Grad-Student. What does it really mean?, <http://eklavyasai.blogspot.com/2012/09/grad-student-what-does-it-really-mean.html> (short link: <http://bit.ly/grad-st> ), Sept. 2012.
21. Open Access Journal Scams, <http://eklavyasai.blogspot.com/2012/09/open-access-journal-scams.html> (short link: <http://bit.ly/open-acce> ), Sept. 2012.

## Science Research

22. Nature April 2014 article - Policy: Free Indian science by Dr. Mathai Joseph et al., <http://eklavyasai.blogspot.com/2014/04/nature-april-2014-article-policy-free.html> (short link: <http://bit.ly/nature-a> ), April 2014.

23. Jayant Narlikar, famous Indian astrophysicist, on Dr. Joseph's Nature article about Indian science policy problems, <http://eklavyasai.blogspot.com/2014/05/jayant-narlikar-famous-indian.html> (short link: <http://bit.ly/jayant-n> ), May 2014.
24. A Hard Look at the Indian Scientific Establishment, <http://eklavyasai.blogspot.com/2013/12/a-hard-look-at-indian-scientific.html> (short link: <http://bit.ly/a-hard-look> ), Dec. 2013.
25. Whistle-blowing on the Internet - Solution for problem of Science Fraud?, <http://eklavyasai.blogspot.com/2014/04/whistle-blowing-on-internet-solution.html> (short link: <http://bit.ly/whistle-b> ), April 2014.

### **Indian Engineering Colleges: General Info. and Graduate Employability**

26. Andhra Pradesh (India) Engineering Colleges - Half Empty!, <http://eklavyasai.blogspot.com/2012/12/andhra-pradesh-india-engineering.html> (short link: <http://bit.ly/andhra-pr> ), Dec. 2012.
27. An "educated guess" analysis of Large no. of Seats in Andhra Pradesh (India) Engg. Colleges Going Empty, <http://eklavyasai.blogspot.com/2012/12/an-educated-guess-analysis-of-large-no.html> (short link: <http://bit.ly/an-educated> ), Dec. 2012.

### **Academic PhD Club & Bureaucratic POWER structure**

28. Mainstream Indian newspaper article on Delhi University's Academic Council - "Everyone is terrified ...", <http://eklavyasai.blogspot.com/2014/07/mainstream-indian-newspaper-article-on.html> (short link: <http://bit.ly/mainstrea> ), July-Sept. 2014.
29. Delhi University Four Year Degree Program Mess! Do Indian university administration mechanisms promote dictatorial tendencies?, <http://eklavyasai.blogspot.com/2014/06/delhi-university-four-year-degree.html> (short link: <http://bit.ly/delhi-uni> ), June 2014.
30. Dictatorial Powers of VC in Indian Central Universities, <http://eklavyasai.blogspot.com/2012/12/dictatorial-powers-of-vc-in-indian.html> (short link: <http://bit.ly/dictato> ), Dec. 2012.
31. UGC scraps Research-linked appraisal for appointments!, <http://eklavyasai.blogspot.com/2013/04/ugc-scraps-research-linked-appraisal.html> (short link: <http://bit.ly/ugc-scr> ), April 2013.
32. Supreme Court rules that AICTE can only advise but not impose sanctions on universities/colleges, <http://eklavyasai.blogspot.com/2013/05/supreme-court-rules-that-aicte-can-only.html> (short link: <http://bit.ly/supreme-co> ), May 2013.
33. Indian Computer Science Academic job can be a DANGEROUS CAREER-WRECK for 'only' graduate Comp.Sc. engineers, <https://eklavyasai.blogspot.com/2017/08/indian-computer-science-academic-job.html> (short link: <http://bit.ly/indian-c> ), Aug. 2017.

### **Academic Reform, Higher Education Bubble, Student Loan Problem & Private Universities**

34. Transparency is the way to differentiate between good and bad deemed universities, <http://eklavyasai.blogspot.com/2014/03/transparency-is-way-to-differentiate.html> (short link: <http://bit.ly/transp-is> ), Mar. 2014.
35. For-profit Corinthian Colleges goes for orderly dissolution of company; One of the largest USA higher education closures; Lessons for India, <http://eklavyasai.blogspot.com/2014/07/for-profit-corinthian-colleges-goes-for.html> (short link: <http://bit.ly/for-profit-c> ), July 2014.
36. Prof. Yashpal Committee: Renovation and Rejuvenation of Higher Education Report (2009) - Extracts related to private higher ed. providers and capitation fee, <http://eklavyasai.blogspot.com/2014/07/prof-yashpal-committee-renovation-and.html> (short link: <http://bit.ly/prof-yash> ), July 2014.
37. Private study (informal data): (Illegal) Donation/Capitation fees in India from KG to PG of Rs. 48,400 crores (US \$ 8 Billion) per year, <http://eklavyasai.blogspot.com/2014/07/private-study-informal-data-illegal.html> (short link: <http://bit.ly/private-st> ), July 2014.

38. Is Brazil blazing a new path for low-cost and high-capacity higher education that countries like India can follow?, <http://eklavyasai.blogspot.com/2014/06/is-brazil-blazing-new-path-for-low-cost.html> (short link: <http://bit.ly/is-brazil> ), posted in June 2014, last updated in July 2014.
39. Tough talking articles about USA Higher Education's 1200+ percent tuition fees increase over past 30 years and Student Debt Trap, <http://eklavyasai.blogspot.com/2014/06/tough-talking-articles-about-usa-higher.html> (short link: <http://bit.ly/tough-talk> ), June 2014.
40. Two Hindu articles on Radically reforming higher education, Student loan procedures and concerns; Aspiring Minds employment linkage to edX MOOC courses, <http://eklavyasai.blogspot.com/2014/06/two-hindu-articles-on-radically.html> (short link: <http://bit.ly/two-hindu> ), June 2014.
41. Recent articles in The Hindu arguing for and against privatisation of (professional) higher education (in India), <http://eklavyasai.blogspot.com/2014/05/recent-articles-in-hindu-arguing-for.html> (short link: <http://bit.ly/recent-a> ), May 2014.
42. Flip-Flop Trauma for 44 Deemed Universities and Lakhs of students, parents, faculty and others associated with them, <http://eklavyasai.blogspot.com/2014/03/flip-flop-trauma-for-44-deemed.html> (short link: <http://bit.ly/flip-flop-t> ), Mar. 2014.
43. Private Deemed Universities - A model for excellence in Indian technical education? But what about its high costs to students, and so, social impact?, <http://eklavyasai.blogspot.com/2014/05/private-deemed-universities-model-for.html> (short link: <http://bit.ly/private-d> ), May 2014.
44. USA Higher Education Bubble? What about Indian Higher Education Bubble?, <http://eklavyasai.blogspot.com/2013/12/usa-higher-education-bubble-what-about.html> (short link: <http://bit.ly/usa-high> ), posted in Dec. 2013, last updated in Jan. 2014.
45. 2013 Milken Institute Panel Discussion - The Future of Higher Education in America, <http://eklavyasai.blogspot.com/2013/12/2013-milken-institute-panel-discussion.html> (short link: <http://bit.ly/2013-milk> ), posted in Dec. 2013, last updated in Jan. 2014.
46. Decline in USA Higher Education - PBS Documentary dated 2005; What about Indian Higher Education?, <http://eklavyasai.blogspot.com/2013/12/decline-in-usa-higher-education-pbs.html> (short link: <http://bit.ly/decline-in> ), Dec. 2013.
47. Think India Education Debate on Indian National English TV Channel, <http://eklavyasai.blogspot.com/2014/01/think-india-education-debate-on-indian.html> (short link: <http://bit.ly/think-india> ), Jan. 2014.
48. For-profit college in California, USA, faces lawsuit from Attorney General for Systematic Deception of Students, <http://eklavyasai.blogspot.com/2014/01/for-profit-college-in-california-usa.html> (short link: <http://bit.ly/for-prof> ), Jan. 2014.
49. Some Serious Dangers of For-Profit Education Schools (Colleges) For Poor and Naive Students, <http://eklavyasai.blogspot.com/2014/01/some-serious-dangers-of-for-profit.html> (short link: <http://bit.ly/some-seri> ), Jan. 2014.
50. Tough Times for US Academia; Lessons for India?, <http://eklavyasai.blogspot.com/2012/09/tough-times-for-us-academia-lessons-for.html> (short link: <http://bit.ly/tough-tim> ), Sept. 2012.
51. Nature News, April 2011, Articles on PhD System Challenges, <http://eklavyasai.blogspot.com/2013/05/nature-news-april-2011-articles-on-phd.html> (short link: <http://bit.ly/nature-n> ), May 2013.
52. USA: The Student Loan Debt Disaster! Update: Income Based Repayment Plan Seems to be a Superb Solution, <http://eklavyasai.blogspot.com/2013/12/the-student-loan-debt-disaster.html> (short link: <http://bit.ly/the-stu> ), posted in Dec. 2013, last updated in Feb. 2014.
53. VIT's flexible credit system seems to give students a lot of choice in courses, teachers & timings, <https://eklavyasai.blogspot.com/2014/11/vits-flexible-credit-system-seems-to.html> (short link: <http://bit.ly/vits-flex> ), Nov. 2014.
54. HRD ministry meetings related to Skill Development in Indian Higher Education including India-US dialogue on it, <https://eklavyasai.blogspot.com/2014/11/hrd-ministry-meetings-related-to-skill.html> (short link: <http://bit.ly/hrd-min> ), Nov. 2014.

55. Indian HRD ministry constitutes council for qualitative reforms in higher education, <https://eklavyasai.blogspot.com/2014/12/indian-hrd-ministry-constitutes-council.html> (short link: <http://bit.ly/indian-h> ), Dec. 2014.
56. TSR Subramanian panel recommends drastically pruning UGC (India's higher education regulator); And perhaps create a new higher regulator?, <http://eklavyasai.blogspot.com/2016/06/tsr-subramanian-panel-recommends.html> (short link: <http://bit.ly/tsr-subra> ), June 2016.

### Online Higher Education; MOOCs

57. Affordable Subject-Wise Certification from Govt. Recognized Academia, <http://eklavyasai.blogspot.com/2011/12/affordable-subject-wise-certification.html> (short link: <http://bit.ly/2w4cWyK> ), posted in Dec. 2011, last updated in June 2013.
58. NPTEL - IIT Madras - offers online course and certification in Programming, Data Structures & Algorithms, <http://eklavyasai.blogspot.com/2014/06/nptel-offers-online-course-and.html> (short link: <http://bit.ly/2uxwYkB> ), June 2014.
59. Georgia Tech, Udacity and AT&T offer Online MS degree in CS under \$7000, <http://eklavyasai.blogspot.com/2013/05/georgia-tech-udacity-and-at-offer.html> (short link: <http://bit.ly/2VpLWnG> ), posted in May 2013, last updated in June 2013.
60. CS & IT: Internet Based Learning, <http://eklavyasai.blogspot.com/2011/10/cs-it-internet-based-learning.html> (short link: <http://bit.ly/2wt9PQU> ), Oct. 2011.
61. Wall Street Journal article on "Higher Education's Online Revolution", <http://eklavyasai.blogspot.com/2012/07/wall-street-journal-article-on-higher.html> (short link: <http://bit.ly/3c7Jrwh> ), July 2012.
62. NYT (New York Times) article on Massively Open Online Courses, <http://eklavyasai.blogspot.com/2012/07/nyt-article-on-massively-open-online.html> (short link: <http://bit.ly/3c55WSJ> ), July 2012.
63. Prof. Sebastain Thrun's experience of teaching 160,000 students AI for free!, <http://eklavyasai.blogspot.com/2012/07/prof-sebastain-thruns-experience-of.html> (short link: <http://bit.ly/prof-seba> ), July 2012.
64. A Princeton Sociology Professor's Online Teaching Experience, <http://eklavyasai.blogspot.com/2012/11/a-princeton-sociology-professors-online.html> (short link: <http://bit.ly/a-princet> ), Nov. 2012.
65. CS & IT Academia: Use Stanford Engineering Everywhere to Teach Programming, <http://eklavyasai.blogspot.com/2011/10/cs-it-academia-teaching-programming.html> (short link: <http://bit.ly/use-stan> ), Oct. 2011.
66. Online Programming Courses: Use MIT's OCW or IIT's NPTEL or Stanford's SEE, <http://eklavyasai.blogspot.com/2011/10/online-programming-courses-use-mits-ocw.html> (short link: <http://bit.ly/online-prog> ), Oct. 2011.
67. How Online Education challenge is shaking up US academia, <http://eklavyasai.blogspot.com/2012/11/how-online-education-challenge-is.html> (short link: <http://bit.ly/how-onlin> ), Nov. 2012.
68. Napster, mp3 music industry disruption and MOOC, <http://eklavyasai.blogspot.com/2012/11/napster-mp3-music-industry-disruption.html> (short link: <http://bit.ly/napster-mp> ), Nov. 2012.
69. The Brains Behind Aakash, the Rs. 3000 (aka \$35) Tablet, <http://eklavyasai.blogspot.com/2011/10/brains-behind-akaash-rs-3000-tablet.html> (short link: <http://bit.ly/brains-b> ), posted in Oct. 2011, last updated in Feb. 2012.
70. Minister HRD provides written response to Parliament question on online learning (ICT) in the country, <https://eklavyasai.blogspot.com/2014/12/minster-hrd-provides-written-response.html> (short link: <http://bit.ly/minster-h> ), Dec. 2014.

71. Sebastian Thrun of Udacity - Hindu interview; Udacity Nanodegrees in India, <http://eklavyasai.blogspot.com/2016/05/sebastian-thrun-of-udacity-hindu.html> (short link: <http://bit.ly/sebastian-t> ), May 2016.

### **Miscellaneous**

72. Dr. S.Ramani's post: Information Technology - Products vs. Services; My comments on it, <http://eklavyasai.blogspot.com/2014/05/information-technology-products-vs.html> (short link: <http://bit.ly/products-vs> ), May 2014.
73. Noted Indian professor of education on young foreign Ph.D. qualified Indians finding it difficult to get good academic jobs in India, and on temporary teacher positions, <https://eklavyasai.blogspot.com/2014/09/noted-indian-professor-of-education-on.html> (short link: <http://bit.ly/noted-in> ), Sept. 2014.
74. PM Modi speech at Digital India event, California, USA on 26th Sept. 2015, <https://eklavyasai.blogspot.com/2015/09/pm-modi-speech-at-digital-india-event.html> (short link: <http://bit.ly/pm-modi-s> ), Sept. 2015.
75. John Chambers of Cisco & chairman-elect of USIBC and PM Narendra Modi speeches at Digital India meet in California, USA, <https://eklavyasai.blogspot.com/2015/09/john-chambers-of-cisco-and-pm-narendra.html> (short link: <http://bit.ly/john-cham> ), Sept. 2015.
76. Business Insider.com article: PhD students have double the risk of developing a psychiatric disorder than the rest of the 'highly educated' population, <https://eklavyasai.blogspot.com/2017/10/business-insidercom-article-phd.html> (short link: <http://bit.ly/business-in> ), 10th Oct. 2017